

PATENT COOPERATION TREATY

PCT

NOTIFICATION OF ELECTION

(PCT Rule 61.2)

From the INTERNATIONAL BUREAU

To:

Assistant Commissioner for Patents
 United States Patent and Trademark
 Office
 Box PCT
 Washington, D.C.20231
 ÉTATS-UNIS D'AMÉRIQUE

in its capacity as elected Office

Date of mailing (day/month/year) 13 March 2000 (13.03.00)	Applicant's or agent's file reference GR 98P2167P
International application No. PCT/DE99/01999	Priority date (day/month/year) 27 July 1998 (27.07.98)
International filing date (day/month/year) 01 July 1999 (01.07.99)	
Applicant BERGER, Michael	

1. The designated Office is hereby notified of its election made:

☒ in the demand filed with the International Preliminary Examining Authority on:

18 February 2000 (18.02.00)

☐ in a notice effecting later election filed with the International Bureau on:2. The election ☒ was☐ was not

made before the expiration of 19 months from the priority date or, where Rule 32 applies, within the time limit under Rule 32.2(b).

The International Bureau of WIPO 34, chemin des Colombettes 1211 Geneva 20, Switzerland Facsimile No.: (41-22) 740.14.35	Authorized officer Kiwa Mpay Telephone No.: (41-22) 338.83.38
--	--

PCT

WELTORGANISATION FÜR GEISTIGES EIGENTUM
Internationales Büro



INTERNATIONALE ANMELDUNG VERÖFFENTLICHT NACH DEM VERTRAG ÜBER DIE
INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS (PCT)

(51) Internationale Patentklassifikation ⁷ : G06F 17/30		A1	(11) Internationale Veröffentlichungsnummer: WO 00/07116 (43) Internationales Veröffentlichungsdatum: 10. Februar 2000 (10.02.00)
(21) Internationales Aktenzeichen: PCT/DE99/01999 (22) Internationales Anmeldedatum: 1. Juli 1999 (01.07.99) (30) Prioritätsdaten: 198 33 778.7 27. Juli 1998 (27.07.98) DE (71) Anmelder (für alle Bestimmungsstaaten ausser US): SIEMENS AKTIENGESELLSCHAFT [DE/DE]; Wittelsbacherplatz 2, D-80333 München (DE). (72) Erfinder; und (75) Erfinder/Anmelder (nur für US): BERGER, Michael [DE/DE]; Am Grenzweg 2, D-85635 Höhenkirchen (DE). (74) Gemeinsamer Vertreter: SIEMENS AKTIENGE- SELLSCHAFT; Postfach 22 16 34, D-80506 München (DE).		(81) Bestimmungsstaaten: US, europäisches Patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE). Veröffentlicht <i>Mit internationalem Recherchenbericht. Vor Ablauf der für Änderungen der Ansprüche zugelassenen Frist; Veröffentlichung wird wiederholt falls Änderungen eintreffen.</i>	
(54) Title: METHOD, ARRANGEMENT AND SET OF A PLURALITY OF ARRANGEMENTS FOR REMEDYING AT LEAST ONE INCONSISTENCY IN A GROUP OF DATABASES WHICH COMPRISES A DATABASE AND AT LEAST ONE COPY DATABASE OF THE DATABASE (54) Bezeichnung: VERFAHREN, ANORDNUNG UND SATZ MEHRERER ANORDNUNGEN ZUR BEHEBUNG MINDESTENS EINER INKONSISTENZ IN EINER DATENBANKMENGE, DIE EINE DATENBANK SOWIE MINDESTENS EINE KOPIEDATENBANK DER DATENBANK AUFWEIST (57) Abstract <p>The aim of the invention is to remedy at least one inconsistency in a group of databases which comprises a database and at least one copy database of the database, said inconsistency arising due to modifications to the data in the database or in the copy database. To this end, the invention provides that at least one part of the operations which can cause an inconsistency is assigned to defined conflict types. A decision set is assigned to each conflict type with which the possible decisions are stated. An inconsistency caused by an operation of the respective conflict type can be remedied using the decisions. Therefore, the inconsistency is remedied by using the decision set. The invention guarantees an error-free remedy of inconsistencies by matching the decision set to the respective situation.</p> (57) Zusammenfassung <p>Zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Daten in der Datenbank oder in der Kopiedatenbank entsteht, ist mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, definierten Konflikttypen zugeordnet. Jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann. Die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben. Eine fehlerfreie Behebung von Inkonsistenzen wird durch Anpassung des Entscheidungssets an die jeweilige Situation gewährleistet.</p>			

LEDIGLICH ZUR INFORMATION

Codes zur Identifizierung von PCT-Vertragsstaaten auf den Kopfbögen der Schriften, die internationale Anmeldungen gemäss dem PCT veröffentlichen.

AL	Albanien	ES	Spanien	LS	Lesotho	SI	Slowenien
AM	Armenien	FI	Finnland	LT	Litauen	SK	Slowakei
AT	Österreich	FR	Frankreich	LU	Luxemburg	SN	Senegal
AU	Australien	GA	Gabus	LV	Letland	SZ	Swasiland
AZ	Aserbaidshan	GB	Vereinigtes Königreich	MC	Monaco	TD	Tschad
BA	Bosnien-Herzegowina	GE	Georgien	MD	Republik Moldau	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagaskar	TJ	Tadschikistan
BE	Belgien	GN	Guinea	MK	Die ehemalige jugoslawische Republik Mazedonien	TM	Turkmenistan
BF	Burkina Faso	GR	Griechenland	ML	Mali	TR	Türkei
BG	Bulgarien	HU	Ungarn	MN	Mongolei	TT	Trinidad und Tobago
BJ	Benin	IE	Irland	MR	Mauretanien	UA	Ukraine
BR	Brasilien	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Island	MX	Mexiko	US	Vereinigte Staaten von Amerika
CA	Kanada	IT	Italien	NE	Niger	UZ	Usbekistan
CF	Zentralafrikanische Republik	JP	Japan	NL	Niederlande	VN	Vietnam
CG	Kongo	KE	Kenia	NO	Norwegen	YU	Jugoslawien
CH	Schweiz	KG	Kirgisistan	NZ	Neuseeland	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Demokratische Volksrepublik Korea	PL	Polen		
CM	Kamerun	KR	Republik Korea	PT	Portugal		
CN	China	KZ	Kasachstan	RO	Rumänien		
CU	Kuba	LC	St. Lucia	RU	Russische Föderation		
CZ	Tschechische Republik	LI	Liechtenstein	SD	Sudan		
DE	Deutschland	LK	Sri Lanka	SE	Schweden		
DK	Dänemark	LR	Liberia	SG	Singapur		
EE	Estland						

Beschreibung

Verfahren, Anordnung und Satz mehrerer Anordnungen zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist

Die Erfindung betrifft ein Verfahren, eine Anordnung sowie einen Satz mehrerer Anordnungen zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist.

Ein solches Verfahren ist aus [1] bekannt.

Bei dem aus [1] bekannten Verfahren kommunizieren Rechner über ein Kommunikationsnetz unter Verwendung eines Kommunikationsprotokolls miteinander.

Unter einem Kommunikationsnetz ist beispielsweise ein Datenetz, ein Funknetz oder auch ein übliches Telefonnetz zu verstehen.

Unter einem Kommunikationsprotokoll ist ein Protokoll zur Festlegung des Datenformats zu verstehen, welches im Rahmen einer Kommunikation zwischen den Rechnern verwendet wird. Ein solches Kommunikationsprotokoll ist beispielsweise das Transport Control Protocol/Internet Protocol (TCP/IP).

Bei dem Verfahren aus [1] sind in einem ersten Rechner eine Datenbank und in jedem weiteren Rechner eine Kopie der Datenbank, im weiteren als Kopiedatenbank bezeichnet, gespeichert.

Die Datenbank bzw. die Kopiedatenbanken werden im Rahmen einer Sitzung von jeweils einem Rechner verändert, d.h. die in der Datenbank bzw. einer Kopiedatenbank enthaltenen Daten bzw. deren Struktur werden verändert.

Unter einer Datenbank ist in diesem Zusammenhang beispielsweise eine hierarchische oder auch eine objektorientierte Datenbank zu verstehen.

5

Eine Datenbank enthält Daten, die gemäß einer vorgegebenen Struktur gespeichert sind und miteinander in Zusammenhang stehen. Jedes Objekt, d.h. jeder Datensatz innerhalb der Datenbank ist üblicherweise über einen Identifikator

10 (Identifizier) eindeutig identifizierbar.

Es kommt vor, daß Änderungen an einer Kopiedatenbank vorgenommen werden, ohne daß dieselbe Änderung auch in der Datenbank selbst erfolgt oder auch umgekehrt.

15

Soll nun aus der jeweiligen Kopiedatenbank und der Datenbank eine konsistente Datenbank erstellt werden, so gilt es, eine durch Hinzufügen, Entfernen oder Ändern der Daten bzw. deren Struktur entstehende Inkonsistenz zu ermitteln und zu beheben.

20

Unter einer Inkonsistenz ist im weiteren jede syntaktische Differenz innerhalb einer Kopiedatenbank bzw. der Datenbank, d.h. alle in den Kopiedatenbanken bzw. der Datenbank auftretenden Abweichungen zwischen den in der Datenbank bzw. einer Kopiedatenbank enthaltenen Datenelementen, ihren Eigenschaften sowie ihren Beziehungen zueinander zu verstehen.

25

In [1] sind verschiedene Möglichkeiten aufgezeigt, um eine solche Inkonsistenz zu beheben.

30

Der Erfindung liegt das Problem zugrunde, ein weiteres Verfahren bzw. eine weitere Vorrichtung zur Behebung von Inkonsistenzen in einer Datenbankmenge anzugeben, mit dem eine möglichst Rechenzeit einsparende Behebung einer Inkonsistenz möglich wird.

35

Das Problem wird durch das Verfahren, durch die Anordnung sowie durch den Satz mehrerer Anordnungen mit den Merkmalen der unabhängigen Patentansprüche gelöst.

- 5 Das Verfahren zur rechnergestützten Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch eine Änderung in der Datenbank und/oder in der Kopiedatenbank entsteht, weist folgende
- 10 Schritte auf:
- a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
 - 15 b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann, und
 - 20 c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.

Die Anordnung zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Daten in der Datenbank oder in der Kopiedatenbank entsteht, weist mindestens einen Prozessor auf, der derart eingerichtet ist, daß folgende Schritte durchführbar sind:

- 30
- a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
 - b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
 - 35

c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.

- 5 Der Satz, mehrerer Anordnungen zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht, weist mehrere Anordnungen auf, deren
- 10 jede mindestens einen Prozessor aufweist, der derart eingerichtet ist, daß folgende Schritte durchführbar sind:
- a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
 - 15 b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
 - c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.
- 20

Die Anordnungen sind miteinander koppelbar.

Durch die Erfindung wird es möglich, eine Inkonsistenz in einer komplexen Datenbank generisch zu lösen.

25

Bevorzugte Weiterbildungen der Erfindung ergeben sich aus den abhängigen Ansprüchen.

In einer bevorzugten Ausgestaltung werden mehrere Inkonsistenzen behoben.

30

Bevorzugt wird in einer weiteren Ausgestaltung jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch

35 mehrere Operationen des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann.

Ferner ist es in einer Weiterbildung vorgesehen, daß die Datenbankmenge mehrere Kopiedatenbanken der Datenbank aufweist.

5 Zur Vereinfachung und somit zur Rechenzeiteinsparung bei der Behebung einer Inkonsistenz ist es in einer Ausgestaltung der Erfindung vorgesehen, daß vor der Behebung jede Inkonsistenz und deren Abhängigkeiten voneinander ermittelt werden.

10 Eine weitere Einsparung an benötigter Rechenzeit zur Behebung mehrerer Inkonsistenzen wird in einer weiteren Ausgestaltung dadurch erreicht, daß das Entscheidungsset mindestens eines Konflikttyps während der Behebung der Inkonsistenzen verändert wird.

15 Dabei erfolgt die Änderung des jeweiligen Entscheidungssets bevorzugt abhängig von Abhängigkeiten der Inkonsistenzen.

20 In einer bevorzugten Ausgestaltung ist es vorgesehen, nach einer vorgebbaren Anzahl behobener Inkonsistenzen die Datenbankmenge auf weitere Inkonsistenzen und deren Abhängigkeit hin zu untersuchen.

25 Die Datenbankmenge enthält bevorzugt in einer Ausgestaltung eine objektorientierte Datenbank.

Das Verfahren kann im Rahmen der objektorientierten Softwareentwicklung oder auch im Rahmen der Erstellung eines strukturierten elektronischen Dokuments eingesetzt werden.

30 Ein Ausführungsbeispiel der Erfindung ist in den Figuren dargestellt und wird im weiteren näher erläutert.

Es zeigen

35 Figur 1 ein Ablaufdiagramm, in dem die Verfahrensschritte des Ausführungsbeispiels dargestellt sind;

Figur 2 eine Skizze, in der Rechner dargestellt sind, die über ein Kommunikationsnetz miteinander verbunden sind;

5 Figur 3 eine Skizze einer Datenbankstruktur;

Figur 4 eine Tabellenübersicht über mögliche Konflikte und Entscheidungssets mit Entscheidungsmöglichkeiten, um die jeweiligen Konflikte zu beheben.

10

Fig.2 zeigt einen ersten Rechner 200 mit einem Speicher 202 und einem Prozessor 203, die jeweils über einen Bus 204 miteinander und mit einer Eingangs-/Ausgangsschnittstelle 201 verbunden sind.

15

Über die Eingangs-/Ausgangsschnittstelle 201 ist der erste Rechner 200 mit einem Bildschirm 205, einer Tastatur 206 sowie einer Computermouse 207 verbunden.

20 Ferner ist der erste Rechner 200 über ein Kommunikationsnetz 260, in dem Beispiel ein ISDN-Netz (Integrated Services Digital Network) mit weiteren Rechnern 210, 220, 230, 240 und 250 verbunden.

25 In dem ersten Rechner 200 ist eine Datenbank 208 gespeichert.

Die weiteren Rechner 210, 220, 230, 240 und 250 weisen jeweils ebenfalls einen Prozessor 213, 223, 233, 243 und 253 sowie jeweils einen Speicher 212, 222, 232, 242 und 252 auf.

30 Jeweils der Prozessor 213, 223, 233, 243 und 253 und der Speicher 212, 222, 232, 242 und 252 sind über jeweils einen Bus 214, 224, 234, 244 und 254 über eine Eingangs-/Ausgangsschnittstelle 211, 221, 231, 241 und 251 mit dem Kommunikationsnetz 260 verbunden. Ferner sind die weiteren
35 Rechner 210, 220, 230, 240 und 250 jeweils mit einem Bildschirm 215, 225, 235, 245 und 255 sowie einer Tastatur 216,

226, 236, 246 und 256 sowie einer Computermouse 217, 227, 237, 247 und 257 verbunden.

Jeweils eine Kopie der Datenbank 208, im weiteren als Kopie-
5 datenbank 218, 228, 238, 248 und 258 bezeichnet, wird von dem ersten Rechner 200 an jeweils einen weiteren Rechner 210, 220, 230, 240 und 250 übermittelt und dort in dessen Speicher 212, 222, 232, 242 und 252 gespeichert.

- 10 Nach Übermittlung der Kopiedatenbanken 218, 228, 238, 248 und 258 unterbrechen die Rechner 200, 210, 220, 230, 240 und 250 die Kommunikation und es erfolgt jeweils unter den Rechnern Rechner 200, 210, 220, 230, 240 und 250 autark eine Änderung, d.h. Entfernung oder Hinzufügung von Daten oder Entfernung
15 oder Hinzufügung von Abhängigkeiten der Daten in einer Kopie- datenbank 218, 228, 238, 248 und 258 bzw. der Datenbank 208.

- Nach Wiederaufnahme der Kommunikation zwischen dem ersten Rechner 200 und den weiteren Rechnern 210, 220, 230, 240 und
20 250 soll eine konsistente Datenbank aus der Datenbank 208 und den Kopiedatenbanken 218, 228, 238, 248 und 258 gebildet werden.

- Zu diesem Zweck ist es erforderlich, jeweils vorgenommene Än-
25 derungen in der Datenbank 208 oder den Kopiedatenbanken fest- zustellen, um somit Inkonsistenzen zwischen den Kopiedaten- banken sowie der Datenbank 208 zu ermitteln, damit die Inkon- sistenzen behoben werden können.

- 30 Unabhängig von der syntaktischen Struktur und der Abhängig- keiten der Datenelemente untereinander kann jedes Datenele- ment beliebig viele Eigenschaften besitzen. Jede Eigenschaft ist dabei von einem bestimmten Eigenschaftstyp und wird durch einen aktuellen Wert repräsentiert. Für alle Eigenschaften
35 wird bezüglich der Wertebereiche die Annahme getroffen, daß alle Werte nur aus Symbolen oder zusammengesetzten Symbolen einer ASCII-Tabelle bestehen dürfen (Ziffern, Zahlen, Buch-

staben, Sonderzeichen, Zeichenketten). Eine Folge solcher Zeichen und Symbole wird nachfolgend als Eintrag bezeichnet. Komplexere Eigenschaften werden bei der Anwendungsmodellierung durch Datenelemente und Beziehungen repräsentiert.

5

Im weiteren werden drei Typen von Eigenschaften bei der syntaktischen Analyse in Abhängigkeit der auf der Eigenschaft ausführbaren Operationen unterschieden:

10 • **Einzelner Wert:**

Ein „Wert“ als Eigenschaftstyp beschreibt einen einzelnen Eintrag, wobei der Eintrag immer in seiner Gesamtheit gesehen und auch so verändert wird. Eine Veränderung der Eigenschaft vom Typ „Wert“ erfolgt dabei immer durch eine vollkommene Ersetzung des Eintrages der Eigenschaft durch einen neuen Eintrag.

15

• **Aufzählung:**

Eine „Aufzählung“ als Eigenschaftstyp beschreibt eine Menge beliebiger Einträge, wobei die Einträge in keiner Relation zueinander stehen und ihrerseits einen einzelnen Wert, eine Aufzählung oder eine geordnete Aufzählung darstellen können. Die einzelnen Einträge können dabei nur einzeln hinzugefügt oder gelöscht werden. Die Eindeutigkeit der Einträge muß bei eventueller Anforderung durch die Anwendung gewährleistet werden. Ein Beispiel für eine Datenstruktur, die diesen Eigenschaftstyp repräsentiert ist eine Hash-Tabelle oder ein Array.

20

25

• **Geordnete Aufzählung:**

Eigenschaften des Typs „geordnete Aufzählung“ beschreiben wie Eigenschaften des einfachen Aufzählungstyps eine Menge beliebiger Einträge. Die Einträge stehen jedoch hier in einer definierten Reihenfolge zueinander, die über einen Index für jeden Eintrag festgelegt ist. Die Festlegung der Indices erfolgt relativ zum Beginn der Aufzählung. Eine Einfüge-Operation mit einzelnen oder mehreren Einträgen bezieht sich deshalb immer auf einen Index. Eine Löschoption kann sich auf einen einzelnen Eintrag mit nur ei-

30

35

nem Index oder auf eine Reihe aufeinanderfolgender Einträge und somit einem Anfangs- und einem Endindex beziehen. Das Kriterium der Reihenfolge wird von einem Anwendungsprogramm definiert und auch die Einhaltung der Reihenfolgekriterien wird von ihm überwacht. Ein Beispiel für diesen Eigenschaftstyp ist eine indizierte Liste mit beliebigen Einträgen (z.B. Textdokument), bei der jede Zeile oder jedes Zeichen einem Eintrag entspricht.

10 Unter einem Datenelement DE ist ein 4-Tupel zu verstehen, welches folgendermaßen definiert ist:

Datenelement

Ein Datenelement DE ist ein 4-Tupel

15

DE def (ID, inforaum, elementtyp, eigenschaften);

- ID ist ein systemweit eindeutiger Identifier
- inforaum \in MIR; wobei MIR eine Menge aller Informationsräume ist
- 20 • elementtyp \in ET; wobei ET eine Menge aller Datenelementtypen ist
- eigenschaften $\subseteq \{(\text{name}, \text{eigenschaftstyp}, \text{wert})\}$:
- name \in MEN, wert \in MEW, eigenschaftstyp \in MET),
- 25 wobei:
- MEN eine Menge aller Eigenschaftsnamen ist und gilt:

$$\forall i \in \{1, \dots, n\}; \forall k \in \{1, \dots, m\}: \text{name}_i \neq \text{name}_k,$$

30 MEW eine Menge aller Eigenschaftswerte ist sowie
MET eine Menge aller Eigenschaftstypen
{„wert“, „aufzählung“, „geordnete aufzählung“} ist.

Ein Informationsraum wird im weiteren folgendermaßen definiert:

35

Informationsraum

Ein Informationsraum IR ist ein 3-Tupel
 IR def (ID, irname, eigentümer, daten)

- ID ist ein systemweit eindeutiger Identifier,
 5 wobei gilt:
 $\forall i \in (1, \dots, n); \forall k \in (1, \dots, n); i \neq k:$
 $IR_i.ID \neq IR_k.ID;$
 wobei MIR eine Menge aller im System vorhandenen Irs ist
 und n deren Anzahl;
- 10 • irname \in MIRN
 wobei gilt:
 $\forall i \in (1, \dots, m); \forall k \in (1, \dots, m); i \neq k:$
 $IR_i.irname \neq IR_k.irname;$
 wobei MIR die Menge aller im System vorhandenen Irs ist und
 15 m deren Anzahl;
 MIRN eine Menge aller möglichen Informationsraumnamen dar-
 stellt;
- eigentümer = N_i mit $N_i \in MN$ oder N_{gi} mit $N_{gi} \in MNG;$
- daten sind die durch eine Nutzergruppe zugreifbaren und dem
 20 IR zugeordneten Daten.

Eine Beziehung zwischen den Datenelementen wird im weiteren
 folgendermaßen definiert:

25 Beziehung zwischen Datenelementen

Eine Beziehung BZ zwischen Datenelementen ist ein 3-Tupel

BZ def (beziehungstyp, name, datenelement1, datenelement2)

- 30 • name \in MBN; wobei MBN eine Menge aller Beziehungsnamen ist
- beziehungstyp \in MBT; wobei MBT eine Menge aller Beziehung-
 stypen
 $\{„ungerichtet“, „logisch“, „nachfolger“, „subsup“\}$ ist
- datenelement1, 2 \in MDE; wobei MDE eine Menge aller Da-
 35 tenelemente ist.

Um Inkonsistenzen zu ermitteln, wird in jedem Rechner 200, 210, 220, 230, 240 und 250 jeweils ein Protokoll über alle an der Datenbank bzw. der jeweiligen Kopiedatenbank vorgenommenen Operation mitgeführt und in Form einer Liste gespeichert.

5

Die gespeicherte Liste wird im weiteren als Historie bezeichnet.

10 Somit ist der Datenbank 208 sowie jeder Kopiedatenbank 218, 298, 238, 248 und 258 jeweils eine Historie zugeordnet.

Diese Situation ist in Fig.3 dargestellt. Fig.3 zeigt die Datenbank 301 mit Objekten 302, 303, 304 und 305 sowie einer Historie 306, die als Einträge 307, 308, 309 Änderungsoperationen gespeichert hat, die seit Unterbrechung der Kommunikation mit den weiteren Rechnern 210, 220, 230, 240 und 250 von dem ersten Rechner 200 an der Datenbank 301 durchgeführt wurden. Die Einträge 307, 308, 309, werden ebenfalls in dem Speicher 202 des ersten Rechners 200 gespeichert.

20

Einer ersten Kopiedatenbank 310 mit Objekten 311, 312, 313 und 314 ist ebenfalls eine Historie 315 mit entsprechenden Änderungsoperationen 316, 317, 318 zugeordnet. Die Kopiedatenbank 310 ist in dem weiteren Rechner 210 gespeichert.

25

Eine zweite Kopiedatenbank 320 mit Objekten 321, 322, 323 und der ihr zugeordneten Historie 325 mit Änderungsoperationen 326, 327, 328 ist in einem weiteren Rechner 220 gespeichert.

30 Zur Bildung der konsistenten Datenbank, d.h. zur Reintegration aller Kopiedatenbanken 218, 228, 238, 248 und 258 mit der Datenbank 208 werden die Historien 315, 325, ... zu dem ersten Rechner 200 über das Kommunikationsnetz 260 übertragen und in dem Speicher 202 des ersten Rechners 200 gespeichert.

35

Zu Beginn der Reintegration, der in Fig.1 durch Schritt 101 beschrieben ist, werden alle Historien der Kopiedatenbanken

12

zu dem ersten Rechner übertragen und dort gespeichert
(Schritt 102).

In einem dritten Schritt (Schritt 103) werden alle im Rahmen
5 der Reintegration zu berücksichtigende Historien 315, 325,
... bestimmt.

Im weiteren werden folgende Änderungsoperationen berücksich-
tigt, mit denen eindeutig die aufgetretenen Inkonsistenzen
10 beschrieben werden.

Im Rahmen dieses Ausführungsbeispiels werden folgende neun
Operationen als Änderungsoperationen berücksichtigt, die im
weiteren in Form eines Pseudo-Programmcodes beschrieben wer-
15 den:

1. Create Element:

Create Element (R(IR), ID, Elementtyp) → R(IR)

```
20 createElement(ir, id, elementtyp) RETURN R(IR)
    BEGIN element := instantiate(elementtyp)
      element.elementname:= id
      R(ir):= insert(R(ir).daten,element)
      R(ir):= add(R(ir).historie,
25      „createElement(id,elementtyp)“)
      return R(ir)
    END
```

Diese Operation erzeugt ein Datenelement vom Datentyp ele-
30 menttyp mit dem Identifikator id in einer Kopiedatenbank
R(ir) innerhalb eines Informationsraums ir, auf den diese
Operation angewendet wird. Dabei erhalten alle Eigenschaften
des neu erzeugten Datenelementes einen vorgeetzten Initiali-
sierungswert. Das neue Element wird nach dessen Initialisie-
35 rung unter dem angegebenen Namen zu den Daten der Kopiedaten-
bank R(ir) und die ausgeführte Operation ohne den Informati-
onsraum als Parameter in die der Kopiedatenbank zugeordneten

13

Historie $R(ir).historie$ hinzugefügt. Die Erzeugung des eindeutigen Identifiers id ist dabei von der die Operation versendenden Anwendung vorzunehmen.

5 2. DeleteElement:

DeleteElement($R(IR), ID$) $\rightarrow R(IR)$

deleteElement(ir, id) RETURN $R(IR)$

```
10        BEGIN element:= select (R(ir).daten, id)
          R(ir):= remove(R(ir).daten, element)
          R(ir):= add(R(ir).historie, "deleteElement(id)")
          return R(ir)
          END
```

15 Diese Operation löscht ein Datenelement mit dem Namen id aus der Kopiedatenbank $R(ir)$ des Informationsraums ir und schreibt die ausgeführte Operation in die der Kopiedatenbank zugeordneten Historie $R(ir).historie$. Alle seit der Instantiierung des Datenelements veränderten Eigenschaften des Datenelementes gehen dabei mit verloren. Die das Element betreffenden Beziehungen bleiben jedoch bestehen. Sollen diese ebenfalls gelöscht werden, so ist das Anwendungsprogramm dafür verantwortlich.

25 3. ChangeEigenschaft:

ChangeEigenschaft($R(IR), ID, Eigenschaftstyp, Wert$) $\rightarrow R(IR)$

```
changeEigenschaft( $ir, id, eigname, neuerWert$ ) RETURN  $R(IR)$ 
30        BEGIN element := select (R(ir).daten, id)
          eigenschaft := select(element.eigenschaften,
          eigname)
          eigenschaft.wert := neuerWert
          R(ir):= add (R(ir).historie, "changeEigenschaft
          (id, eigname, neuerWert)")
35        return R(ir)
          END
```

14

Diese Operation setzt in der Kopiedatenbank $R(ir)$ des Informationsraums ir den Wert der Eigenschaft **eigname** des Datenelementes mit dem Identifier **id** auf den Wert **neuerWert** und schreibt die ausgeführte Operation in die der Kopiedatenbank zugeordneten Historie $R(ir).historie$.

4. ChangeEigenschaftAdd:

ChangeEigenschaftAdd($R(IR)$, **ID**, **Eigenschaftstyp**, **Eintrag**) $\rightarrow R(IR)$

```

10  changeEigenschaftAdd(ir,id,eigname,neuerEintrag) RETURN  $R(IR)$ 
      BEGIN element := select ( $R(ir).daten$ , id)
        eigenschaft := select (element.eigenschaften,
        eigname)
        add (eigenschaft.wert, neuerEintrag)
15     $R(ir)$  := add( $R(ir).historie$ ,
        „changeEigenschaftAdd (id, eigname, neuerEintrag)“)
        return  $R(ir)$ 
      END
20
```

Diese Operation ist für Eigenschaften des Typs „aufzählung“. Die Operation fügt in der Kopiedatenbank $R(ir)$ des Informationsraums ir im Wert der Eigenschaft **eigname** des Datenelementes mit dem Identifier **id** am Ende der Aufzählung einen neuen Eintrag **neuerEintrag** hinzu. Anschließend wird die ausgeführte Operation in die der Kopiedatenbank zugeordneten Historie $R(ir).historie$ gespeichert.

15

5. ChangeEigenschaftDel:

ChangeEigenschaftDel(R(IR), ID, Eigenschaftstyp, Index,
Eintrag) → R(IR)

```
5  changeEigenschaftDel(ir, id, eigname, index, alterEintrag)
    RETURN R(IR)
    BEGIN element := select (R(ir).daten, id)
      eigenschaft := select (element.eigenschaften,
10      eigname)
      del(eigenschaft.wert, alterEintrag)
      R(ir) := add(R(ir).historie,
        „changeEigenschaftDel(id, eigname, index, al-
        terEintrag)“)
      return R(ir)
15      END
```

Diese Operation ist für Eigenschaften des Typs „aufzählung“. Die Operation löscht in der Kopiedatenbank R(ir) des Informationsraums ir im Wert der Eigenschaft eigname des Datenelementes mit dem Identifier id den ersten in der Aufzählung auftretenden Eintrag alterEintrag. Danach wird die ausgeführte Operation in der der Kopiedatenbank zugeordneten Historie R(ir).historie gespeichert.

6. ChangeEigenschaftInsert:

ChangeEigenschaftInsert(R(IR), ID, Eigenschaftstyp, Index,
Anzahl, Einträge) → R(IR)

```

5  changeEigenschaftInsert(ir, id, eigname, index, anzahl,
    neueEinträge) RETURN R(IR)
    BEGIN element := select (R(ir).daten, id)
      eigenschaft := select (element.eigenschaften,
10      eigname)
      for (i=0, i < anzahl, i++){
        incrIndex (eigenschaft.wert.einträge, ind >= index)
        insert (eigenschaft.wert.index, neueEinträge.
15      (anzahl -i))
      R(ir) := add (R(ir).historie,
        „changeEigenschaftInsert (id, eigname, index, an-
        zahl, neueEinträge)“)
      return R(ir)
    END
20
```

Eine Eigenschaft, auf die diese Operation angewendet werden kann, ist vom Typ „geordnete aufzählung“. Diese Operation fügt in der Kopiedatenbank R(ir) des Informationsraumes ir im Wert der Eigenschaft eigname des Datenelementes mit dem Identifier id ab der Position Index in der geordneten Aufzählung Eigenschaft.Wert eine Anzahl anzahl neuer Einträge neueEinträge.i ein. Alle Einträge der geordneten Aufzählung eigenschaft.Wert mit einem gleichen oder größeren Index als index wird index um den Wert anzahl erhöht. Danach wird die ausgeführte Operation in der der jeweilige Kopiedatenbank zugeordneten Historie R(ir).historie, gespeichert.

7. ChangeEigenschaftRemove:

ChangeEigenschaftRemove(R(IR), ID, Eigenschaftstyp, Index,
Anzahl, Einträge) → R(IR)

```

5  changeEigenschaftRemove(ir, id, eigname, index, anzahl,
    alteEinträge) RETURN R(IR)
    BEGIN element := select (R(ir).daten,id)
      eigenschaft := select (element.eigenschaften,
10      eigname)
      for (i = 0, i < anzahl, i++){
        remove (eigenschaft.wert.index, alteEinträge.
          (i+1))
        decrIndex (eigenschaft.wert.einträge, ind > in-
15        dex)}
        R(ir) := add(R(ir).historie,
          „changeEigenschaftRemove(id, eigname, index, an-
          zahl, alteEinträge“)
        return R(ir)

```

END

20

Eine Eigenschaft, auf die diese Operation angewendet werden kann, ist vom Typ „geordnete aufzählung“. Diese Operation löscht in der Kopiedatenbank R(ir) des Informationsraumes ir im Wert der Eigenschaft eigname des Datenelementes mit dem

25 Identifizier id ab der Position Index in der geordneten Aufzählung Eigenschaft.Wert die Einträge alteEinträge. Alle Einträge mit größerem Index als (index + anzahl) werden um den Wert anzahl in ihrem Index verringert. Danach wird die ausgeführte Operation in der jeweiligen Kopiedatenbank zugeordneten

30 Historie R(ir).historie gespeichert.

8. CreateBeziehung:**CreateBeziehung**(R(IR), Name, Beziehungstyp, ID1, ID2) → R(IR)

```
createBeziehung(ir, name, beztyp, fromid, toid, toidir)
5      RETURN R(IR)
      BEGIN beziehung := instantiate (beztyp)
          beziehung.name := name
          beziehung.datenelement1 := fromid
          beziehung.datenelement2 := toid
10      beziehung.datenelement2.ir := toidir
          R(ir) := insert (R(ir).daten, beziehung)
          R(ir) := add (R(ir).historie,
            „createBeziehung (name, beztyp, fromid, toid,
              toidir)“)
15      return R(ir)
      END
```

Diese Operation erzeugt eine Beziehung des Typs **beztyp** zwischen den Datenelementen mit den Identifiern **fromid** and **toid** unter dem Namen **name** und fügt die neue Beziehung zu den Daten der Kopiedatenbank **R(ir)** des Informationsraums **ir** hinzu. Danach wird die ausgeführte Operation in der der Kopiedatenbank zugeordneten Historie **R(ir).historie** gespeichert. Es wird für alle Beziehungen angenommen, daß es pro vergebenen Beziehungsnamen nur eine einzige Beziehung des gleichen Typs zwischen zwei Datenelementen gibt. Sind mehrere Beziehungen des gleichen Typs unter dem gleichen Namen zwischen zwei Datenelementen notwendig, so sind auch für Beziehungen Identifier einzuführen. Für die Mehrzahl der Anwendungen reicht die getroffene Annahme jedoch aus. Die Angabe des Informationsraumes des Zieldatenelementes ist nur im Falle einer logisch extern gerichteten Beziehung notwendig.

9. DeleteBeziehung:

DeleteBeziehung(R(IR), Name, Beziehungstyp, ID1, ID2) → R(IR)

deleteBeziehung(ir, name, beztyp, fromid, toid.toidir)

```

5         RETURN R(IR)
        BEGIN beziehung := select (R(ir).daten, name, beztyp,
                                   fromid, toid)
        R(ir) := remove (R(ir).daten, beziehung)
        R(ir) := add (R(ir).historie,
10         „deleteBeziehung (name, beztyp, fromid,
        toid.toidir)“)
        return R(ir)
        END

```

- 15 Diese Operation löscht eine Beziehung des Typs **beztyp** zwischen den Datenelementen mit den Identifiern **fromid** und **toid** unter dem Name **name** aus der Kopiedatenbank **R(ir)** des Informationsraumes **ir**. Danach wird die ausgeführte Operation in der der Kopiedatenbank zugeordneten Historie **R(ir).historie** gespeichert. Die Angabe des Informationsraums des Zieldatenelements ist nur für die Beziehungen vom Typ logisch extern gerichtet notwendig.
- 20

- In einem weiteren Schritt werden alle Konflikte, Abhängigkeiten, Anomalien, Pseudo-Anomalien, sowie Einschränkungen durch Abhängigkeiten erkannt (Schritt 103).
- 25

- Unter einem Konflikt ist die kleinste entscheidbare Menge von syntaktisch nur einseitig auftretenden Operationen, die eine Inkonsistenz eindeutig beschreiben und einem Nutzer oder dem System sinnvoll präsentiert und von ihm behoben (entschieden) werden können, zu verstehen.
- 30

- Jeder Konflikt ist als Ganzes zu erkennen und durch eine einzige Entscheidung während der Reintegration lösbar.
- 35

Mögliche Konflikte werden anschließend abhängig von der Datenstruktur und den in der Historie vorkommenden Operationen definiert.

- 5 Es werden harmlose Konflikte und kritische Konflikte unterschieden.

Harmlose Konflikte (HK) beinhalten nur Operationen, die Veränderungen auf einer Kopiedatenbank beschreiben. Es gibt in
10 diesem Fall somit nur einen Nutzer, der eine Änderung an dem Teil der Datenstruktur bzw. der Kopiedatenbank oder auch der Datenbank selbst wünscht und durchführt. Die vorgenommenen Operationen ergänzen sich somit. Abhängig davon, welcher Kopiedatenbank die Operation bzw. Operationen zuzuordnen sind,
15 können allgemein bei Vorhandensein der Kopiedatenbanken von Benutzern A und B harmlose Konflikte mit Operationen auf der Kopiedatenbank eines ersten Benutzers A als HKA und harmlose Konflikte mit Operationen auf der Kopiedatenbank eines zweiten Benutzers B als HKB bezeichnet werden.

20

Kritische Konflikte (KK) dagegen enthalten beidseitige Änderungen zum gleichen Teil der Datenstruktur und stellen konträre Ansichten der Benutzer über den letztendlichen Zustand bestimmter Daten innerhalb der Datenbank bzw. Kopiedatenbanken dar. Dabei kann ein kritischer Konflikt auch durch unterschiedliche Operationen auf zwei Kopiedatenbanken definiert
25 werden. In diesen Fällen wird zwischen einem kritischen Konflikt KKA und einem kritischen Konflikt KKB unterschieden.

- 30 Formal wird ein Konflikt wie folgt definiert:

Konflikt:

Ein Konflikt K zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 6-Tupel

5 K(EHA, EHB, GH) def

(id, ktyp, operationenEHA, operationenEHB,
operationenGH, entscheidungseinschr);

- id ist ein systemweit eindeutiger Identifier (siehe auch
10 die Definition eines Datenelements)
- ktyp \in {HK1A, ..., HK11A, HK1B, ..., HK11B, KK1, KK2,
KK3A, KK3B, KK4, KK5A, KK5B, KK6, KK7, KK8A, KK8B}
- operationenEHA \in EHA.operationen;
- operationenEHB \in EHB.operationen;
- 15 • operationenGH \in GH.operationen;
- entscheidungseinschr \subseteq MENTktyp; wobei MENTktyp eine Menge
aller möglichen Entscheidungen für einen Konflikt vom Typ
ktyp ist.

20 Die Konflikte sind in Fig.4 dargestellt.

1. Erster Harmloser Konflikt HK1:

HK1 = (createElement/ --)

Es liegt eine Erzeugungsoperation eines Datenelements

25 createElement(id, elementtyp) in nur einer Historie vor.

HK1A = createElement(id, elementtyp) \in EHA \vee

HK1B = createElement(id, elementtyp) \in EHB.

2. Zweiter Harmloser Konflikt HK2:

30 HK2 = (deleteElement/ --)

Es liegt eine Löschoperation zu einem Datenelement

deleteelement(id, elementtyp) in nur einer Historie vor.

HK2A = deleteElement(id, elementtyp) \in EHA \vee

HK2B = deleteElement(id, elementtyp) \in EHB.

35

3. Dritter Harmloser Konflikt HK3:

HK3 = (createBeziehung/ --)

Es liegt eine Erzeugungsoperation einer Beziehung

5 createBeziehung (beztyp, bname, id1, id2) in nur einer Historie vor.

HK3A = createBeziehung(beztyp, bname, id1, id2) ∈ EHA ∨

HK3B = createBeziehung(beztyp, bname, id1, id2) ∈ EHB.

4. Vierter Harmloser Konflikt HK4:

10 HK4 = (deleteBeziehung/ --)

Es liegt eine Löschoption einer Beziehung

deleteBeziehung(beztyp, bname, id1, id2) in nur einer Historie vor.

HK4A = deleteBeziehung(beztyp, bname, id1, id2) ∈ EHA ∨

15 HK4B = deleteBeziehung(beztyp, bname, id1, id2) ∈ EHB.

5. Fünfter Harmloser Konflikt HK5:

HK5 = (deleteBeziehung12, createBeziehung 13/ --)

Es liegt eine Löschoption einer Beziehung

20 deleteBeziehung(beztyp, bname id1, id2) und eine nachfolgende Erzeugungsoperation der Beziehung
createBeziehung(beztyp, bname, id1, id3) vom gleichen Quell-
datenelement zu einem anderen Zieldatenelement nur einer Historie vor.

25 HK5A = [deleteBeziehung(beztyp, bname, id1, id2),
createBeziehung(beztyp, bname, id1, id3)] ∈ EHA ∨

HK5B = [deleteBeziehung(beztyp, bname, id1, id2),
createBeziehung(beztyp, bname, id1, id3)] ∈ EHB.

30 **6. Sechster Harmloser Konflikt HK6:**

HK6 = (changeEigenschaft/ --)

Es liegt eine Änderungsoperation

changeEigenschaft(id, name, wertneu, wertalt) zu einer Eigenschaft vom Typ „Wert“ in nur einer Historie vor.

35 HK6A = changeEigenschaft(id, name, wert1, wert0) ∈ EHA ∨

HK6B = changeEigenschaft(id, name, wert1, wert0) ∈ EHB.

7. Siebter Harmloser Konflikt HK7:

HK7 = (n x changeEigenschaftAdd/ --)

Es liegen n (n ist Element der natürlichen Zahlen und $n > 0$)

Einfügeoperationen changeEigenschaftAdd(id, name, eintrag)

- 5 mit dem gleichen Eintrag eintrag zur gleichen Eigenschaft vom Typ „aufzählung“ eines Datenelements in einer Historie vor, wobei es in einer anderen Historie keine Löschope-
 10 rationen mit dem gleichen Eintrag zur gleichen Eigenschaft des Datenelements gibt. Die beschriebene Inkonsistenz besteht darin, daß in der Kopiedatenbank mit den Erzeugungsoperationen in der der Kopiedatenbank zugeordneten Historie n Einträge der Art eintrag mehr vorhanden sind als in der jeweils anderen Kopie-
 datenbank.

HK7A = n mal changeEigenschaftAdd(id,name,eintrag) \in EHA \vee

- 15 HK7B = n mal changeEigenschaftAdd(id,name,eintrag) \in EHB.

8. Achter Harmloser Konflikt HK8:

HK8 = (n x changeEigenschaftDel/ --)

Es liegen n (n ist Element der natürlichen Zahlen) Löschope-

- 20 rationen changeEigenschaftDel(id, name, eintrag) mit dem gleichen Eintrag eintrag zu einer Eigenschaft vom Typ „aufzählung“ eines Datenelements in einer Historie vor, wobei in einer anderen Historie keine Einfügeoperation mit dem gleichen Eintrag zu der Eigenschaft des Datenelements vor-
 25 liegt. Die beschriebene Inkonsistenz besteht darin, daß in der Kopiedatenbank, in dessen Historie die Löschooperationen stehen, die n Einträge der Art eintrag weniger vorhanden sind, als in der jeweils anderen Kopiedatenbank.

HK8A = n mal changeEigenschaftDel(id,name,eintrag) \in EHA \vee

- 30 HK8B = n mal changeEigenschaftDel(id,name,eintrag) \in EHB.

9. Neunter Harmloser Konflikt HK9:

HK9 = (changeEigenschaftInsert/ --)

Es liegt eine Einfügeoperation

- 35 changeEigenschaftInsert(id, name, index1, 1, eintrag1) zu einem Index mit einem einzelnen Eintrag zu einer Eigenschaft vom Typ „geordnete aufzählung“ eines Datenelementes in nur

einer Historie vor, wobei in einer anderen Historie keine Einfügeoperationen mit einem anderen Eintrag zu nachrechenbarem gleichen Index zu der gleichen Eigenschaft des Datenelementes vorliegt.

5 HK9A = changeEigenschaftInsert(id,name,index,1,eintrag)
 ∈ EHA √
 HK9B = changeEigenschaftInsert(id,name,index,1,eintrag)
 ∈ EHB.

10 10. Zehnter Harmloser Konflikt HK10:

 HK10 = (changeEigenschaftRemove/ --)

 Es liegt eine Löschoperation

 changeEigenschaftRemove(id, name, index1, 1, eintrag1) zu einem Index mit einem einzelnen Eintrag zu einer Eigenschaft

15 vom Typ „geordnete aufzählung“ eines Datenelements in nur einer Historie vor.

 HK10A = changeEigenschaftIRemove(id,name,index,1,eintrag)
 ∈ EHA √

 HK10B = changeEigenschaftIRemove(id,name,index,1,eintrag)

20 ∈ EHB.

11. Elfter Harmloser Konflikt HK11:

HK11 = (changeEigenschaftRemove, changeEigenschaftInsert/--)

Es liegt eine Löschoperation

changeEigenschaftRemove(id, name, index1, 1, eintrag1) zu einem Index index mit einem einzelnen Eintrag zu einer Eigenschaft vom Typ „geordnete aufzählung“ eines Datenelements und eine nachfolgende Erzeugungsoperation

changeEigenschaftInsert (id, name, index1, 1, eintrag2) eines Eintrages zum gleichen Index der gleichen Eigenschaft des gleichen Datenelements in nur einer Historie vor.

HK11A = [changeEigenschaftIRemove(id, name, index, 1, eintrag1),
changeEigenschaftIInsert(id, name, index, 1, eintrag2)]
∈ EHA ∨

HK11B = [changeEigenschaftIRemove(id, name, index, 1, eintrag1),
changeEigenschaftIInsert(id, name, index, 1, eintrag2)]
∈ EHB.

Im weiteren wird eine Übersicht über kritische Konflikte (KK) d.h. Operationen in mehreren Historien, gegeben:

1. Erster Kritischer Konflikt KK1:

KK1 = (createBeziehung12/ createBeziehung13)

Es liegt eine Erzeugung einer Beziehung

createBeziehung(beztyp, bname, id1, id2) in einer Historie

vor, wobei in einer anderen Historie eine Erzeugung

createBeziehung(beztyp, bname, id1, id3) der gleichen Beziehung (beztyp, bname) vom gleichen Quelldatenelement ausgehend aber zu einem anderen Zieldatenelement existiert.

KK1 = createBeziehung(beztyp, bname, id1, id2) ∈ EHA ∧

createBeziehung(beztyp, bname, id1, id3) ∈ EHB.

2. Zweiter Kritischer Konflikt KK2:

KK2 = (deleteBeziehung12, createBeziehung13 /
deleteBeziehung12, createBeziehung14)

Die unterschiedliche Änderung einer Beziehung kann in den Historien wie der erste kritische Konflikt KK1 erkannt werden. Zusätzlich liegt jedoch in einer gemeinsamen Historie GH eine

Löschung deleteBeziehung(beztyp, bname, id1, id2) der gemeinsamen Beziehung (bname, beztyp) vom gleichen Quelldatenelement, aber zu einem anderen Zieldatenelement vor.

KK2 = deleteBeziehung(beztyp, bname, id1, id2) \in GH \wedge
 5 createBeziehung(beztyp, bname, id1, id3) \in EHA \wedge
 createBeziehung(beztyp, bname, id1, id4) \in EHB.

3. Dritter Kritischer Konflikt KK3:

KK3 = (deleteBeziehung12, createBeziehung13/
 10 deleteBeziehung12)

Die einseitige Veränderung und anderseitige Löschung einer Beziehung kann in den Historien wie der dritte harmlose Konflikt HK3 durch eine Operation

createBeziehung(beztyp, bname, id1, id3) erkannt werden.
 15 Zusätzlich liegt jedoch in der gemeinsamen Historie eine
 Löschung

deleteBeziehung (beztyp, bname, id1, id2) der gemeinsamen Beziehung (bname, beztyp) vom gleichen Quelldatenelement, aber zum letzten gemeinsamen Zieldatenelement vor.

20 KK3A = deleteBeziehung(beztyp, bname, id1, id2) \in GH \wedge
 createBeziehung(beztyp, bname, id1, id3) \in EHA;
 KK3B = deleteBeziehung(beztyp, bname, id1, id2) \in GH \wedge
 createBeziehung(beztyp, bname, id1, id3) \in EHB.

25 4. Vierter Kritischer Konflikt KK4:

KK4 = (changeEigenschaft/ changeEigenschaft)

Es liegt eine Änderungsoperation

changeEigenschaft (id, name, wertneul, wertalt) zu einer Eigenschaft des Typs „wert“ eines Datenelements in einer Historie vor, wobei in einer anderen Historie zur gleichen Eigenschaft des Datenelements eine andere Änderungsoperation existiert.
 30

KK4 = changeEigenschaft(id, name, wertneul, wertalt) \in EHA \wedge
 changeEigenschaft(id, name, wertneu2, wertalt) \in EHB.

35

5. Fünfter Kritischer Konflikt KK5:

KK5 = (n changeEigenschaftAdd/ m changeEigenschaftDel)

Es liegen n (n ist Element der natürlichen Zahlen) gleiche Operationen der Art

- 5 changeEigenschaftAdd(id, name, eintrag) mit dem gleichen Eintrag zu einer Eigenschaft vom Typ „aufzählung“ eines Datenelements in einer Historie vor, wobei in einer anderen Historie m (m ist Element der natürlichen Zahlen) gleiche Operationen der Art
- 10 changeEigenschaftDel(id, name, eintrag) mit dem gleichen Eintrag zur gleichen Eigenschaft des Datenelements vorliegen. Die beschriebene Inkonsistenz besteht darin, daß in der Kopiedatenbank mit den Erzeugungsoperationen in der der Kopiedatenbank zugeordneten Historie n + m gleiche Einträge eintrag mehr vorhanden sind als in der anderen Kopiedatenbank.
- 15 Um eine exakte Aussage über das Auftreten der Operationen treffen zu können, wird bei dem fünften kritischen Konflikt KK5 zwischen einem fünften kritischen Konflikt erster Art KK5A und einem fünften kritischen Konflikt zweiter Art KK5B unterschieden. Die Zuordnung erfolgt dabei über die Erzeugungsoperationen.
- 20
- KK5A = (n changeEigenschaftAdd(id,name,eintrag) \in EHA \wedge
m changeEigenschaftDel(id,name,eintrag) \in EHB \vee
- KK5B = (m changeEigenschaftDel(id,name,eintrag) \in EHA;
25 n changeEigenschaftAdd(id,name,eintrag) \in EHB.

6. Sechster Kritischer Konflikt KK6:

KK6 = (changeEigenschaftInsert / changeEigenschaftInsert)

Es liegt eine Einfügeoperation

changeEigenschaftInsert(id, name, index1, 1, eintrag1) zu ei-

- 5 nem Index mit einem einzelnen Eintrag zu einer Eigenschaft vom Typ „geordnete aufzählung“ eines Datenelements in einer Historie vor, wobei in einer anderen Historie eine Einfügeoperation mit einem anderen Eintrag zum nachrechenbar gleichen Index der Eigenschaft des Datenelements vorliegt.

- 10 KK6 = changeEigenschaftInsert(id, name, index1, 1, eintrag1)
 \in EHA \wedge
 changeEigenschaftInsert(id, name, index2, 1, eintrag2)
 \in EHB.

15 7. Siebter Kritischer Konflikt KK7:

KK7 = (changeEigRemove, changeEigInsert /
 changeEigRemove, changeEigInsert)

Die beidseitig unterschiedliche Änderung eines einzelnen Eintrages an einem gemeinsamen Index index zu einer Eigenschaft

- 20 vom Typ „geordnete aufzählung“ eines Datenelementes ist in den Historien wie der sechste kritische Konflikt KK6 erkennbar. Zusätzlich liegt jedoch in der gemeinsamen Historie eine changeEigenschaftRemove(id, name, index1, 1, eintrag1)-Operation des letzten gemeinsamen Eintrages am nachrechenbar
 25 gleichen Index vor.

- KK7 = changeEigenschaftRemove(id, name, index1, 1, eintrag1)
 \in GH \wedge
 changeEigenschaftInsert(id, name, index2, 1, eintrag2)
 \in EHA \wedge
 30 changeEigenschaftInsert(id, name, index3, 1, eintrag3)
 \in EHB.

8. Achter Kritischer Konflikt KK8:

KK8 = (changeEigRemove, changeEigInsert / changeEigRemove)

Die einseitige Änderung und anderseitige Löschung eines einzelnen Eintrages an einem gemeinsamen Index `index` zu einer

- 5 Eigenschaft vom Typ „geordnete aufzählung“ eines Datenelements in den Historien ist wie ein zehnter harmloser Konflikt HK10 erkennbar. Zusätzlich liegt jedoch in der gemeinsamen Historie eine
- 10 `changeEigenschaftRemove(id, name, index1, 1, eintrag1)`-Operation des letzten gemeinsamen Eintrages am nachrechenbar gleichen Index vor.

KK8A = `changeEigenschaftRemove(id, name, index1, 1, eintrag1)`

∈ GH ∧

`changeEigenschaftInsert(id, name, index2, 1, eintrag2)`

- 15 ∈ EHA;

KK8B = `changeEigenschaftRemove(id, name, index1, 1, eintrag1)`

∈ GH ∧

`changeEigenschaftInsert(id, name, index2, 1, eintrag2)`

∈ EHB.

20

Die in den Historien gespeicherten Operationen beschreiben die autonom veränderten Datenbereiche direkt und werden zur Beschreibung sowie zur im weiteren beschriebenen Behebung der Inkonsistenzen verwendet.

25

Zur Erkennung der Inkonsistenzen werden jeweils zwei Historien miteinander verglichen.

- 30 Die Erkennung der Inkonsistenzen erfolgt zu Beginn des Verfahrens, vor der eigentlichen Reintegration.

Die Suche nach den vorhandenen Inkonsistenzen in den Kopiedatenbanken durch die Suche nach Konfliktoperationen erfolgt in den im weiteren beschriebenen drei Schritten.

35

- In einem ersten Schritt werden die zwei miteinander zu vergleichenden Historien von den Kopiedatenbanken bzw. der Da-

tenbank durchlaufen, die miteinander abgeglichen werden sollen. Alle Operationen der Historien werden auf jeder Seite getrennt jeweils einer der oben beschriebenen neuen Operationsmengen (createElement-Operationen, deleteElement-Operationen, createBeziehung-Operationen, delete-Beziehung-Operationen, changeEigenschaft-Operationen, changeEigenschaftAdd-Operationen, changeEigenschaftDel-Operationen, changeEigenschaftInsert-Operationen und ChangeEigenschaftRemove-Operationen) zugeordnet.

10

- In einem zweiten Schritt wird für jeden oben beschriebenen Konflikttyp HK1A, ..., HK11A, HK1B, ..., HK11B, KK1, KK2, KK3A, KK3B, KK4, KK5A, KK5B, KK6, KK7, KK8A, KK8B jeweils ein Konfliktregister KR angelegt. Dabei wird gewährleistet, daß alle Konflikte, bei denen Operationen aus beiden Historien zum jeweiligen Konflikt beitragen, nicht doppelt erkannt und doppelt in dem jeweiligen Konfliktregister KR abgelegt werden. Danach werden entsprechend den Definitionen der Konflikttypen, wie oben beschrieben, beginnend mit dem ersten harmlosen Konflikt HK1A in der Historie des ersten Benutzers A, die gerade gebildeten Operationsmengen durchsucht. Wurde ein Konflikt ermittelt, so wird der Konflikt im Konfliktregister KR des entsprechenden Konflikttyps abgelegt, beispielsweise wird der erste harmlose Konflikt HK1 in der Kopiedatenbank des ersten Benutzers A in einem Konfliktregister KR_HK1A abgelegt.

20

25

- Ist die Suche und Speicherung der Konflikte erfolgt, werden in einem dritten Schritt die zu Beginn erstellten Operationsmengen wieder gelöscht. Die nachfolgende Behebung der Inkonsistenzen beruht auf den in den Konfliktregistern KR abgelegten Konflikten und deren Operationen.

30

Ein Konfliktregister KR ist wie folgt definiert:

35

Konfliktregister KR:

Ein Konfliktregister KR zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 2-Tupel

5 KR(EH_A, EH_B, GH) def (krtyp, konfliktids)

• krtyp \in {KA_HK1A, ..., KA_HK11A, KA_HK1B, ..., KA_HK11B, KA_KK1, KA_KK2, KA_KK3A, KA_KK3B, KA_KK4, KA_KK5A, KA_KK5B, KA_KK6, KA_KK7, KA_KK8A, KA_KK8B}

10 • konfliktids sind Identifier aller dem Konfliktregister KR zugeordneten Konflikte K(EHA, EHB, GH), wobei K.typ dem jeweiligen Konfliktarraytyp KR.krtyp zuzuordnen ist.

15 Eine Anomalie liegt dann vor, wenn zwei Datenelemente in beiden Kopiedatenbanken vor und nach der Teilung existieren und diese nach der Teilung durch eine gerichtete Beziehung vom gleichen Typ beztyp und mit gleichem Namen bname, jedoch mit vertauschtem Quelldatenelement und Zieldatenelement verbunden sind. Während der Reintegration muß mindestens eine dieser
20 Beziehungen abgelehnt oder müssen beide verändert werden.

Unter einer gerichteten Beziehung ist eine Beziehung zu verstehen, die von einem Zieldatenelement zu einem Quelldatenelement gerichtet ist.

25

Eine Anomalie ist wie folgt definiert:

Anomalie:

Eine Anomalie AM zweier Konflikte K1 (EHA, EHB, GH) und K2 (EHA, EHB, GH) ist ein 4-Tupel

5 AM(K1, K2) def (id, amtyp, kid1, kid2)

- id ist ein systemweit eindeutiger Identifier (siehe auch Definition eines Datenelementes)
- amtyp \in {Anomalie1A, ..., Anomalie16AB}
- 10 • kid1 = K1.id
- kid2 = K2.id.

Ein Anomalieregister AMR zweier Historien ist wie folgt definiert:

15

Anomalieregister AMR:

Ein Anomalieregister AMR zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 1-Tupel

20 AMR (EHA, EHB, GH) def (anomalieids)

- anomalieids sind die Identifier aller Anomalien der Historien EHA und EHB und der gemeinsamen Historie GH.
- 25 Pseudo-Anomalien beschreiben Situationen, in denen das Entstehen einer Anomalie aus vorliegenden Konflikten nur durch eine gezielte Minimierung der Entscheidungsmöglichkeiten der Konflikte vermieden werden kann.
- 30 Eine Pseudo-Anomalie ist, wie im folgenden dargestellt, definiert:

Pseudo-Anomalie PAM:

Eine Pseudo-Anomalie PAM zweier Konflikte K1(EHA, EHB, GH) und K1(EHA, EHB, GH) ist ein 4-Tupel

5 PAM(K1, K2) def (id, pamtyp, kid1, kid2)

- id ist ein systemweit eindeutiger Identifier (siehe auch Definition eines Datenelementes)
- pamtyp \in {Pseudo-Anomalie1A, ..., Pseudo-Anomalie32AB}
- 10 • kid1 = K1.id
- kid2 = K2.id

Ein Pseudo-Anomalieregister PAMR ist wie folgt definiert:

15 **Pseudo-Anomalieregister PAMR:**

Ein Pseudo-Anomalieregister PAMR zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 1-Tupel

PAMR(EHA, EHB, GH) def (Pseudo-Anomalieids)

20

- Pseudo-Anomalieids sind die Identifier aller Pseudo-Anomalien der Historien EHA und EHB sowie der gemeinsamen Historie GH.

25 Nach Ermittlung der Konflikte wird jeder ermittelte Konflikt jeweils durch eine einzelne Entscheidung gelöst. Der Konfliktlösungsprozeß besteht somit aus einer Sequenz von Konfliktlösungsentscheidungen.

30 Die Konfliktlösung ist in Fig.1 mit Schritt 104 bezeichnet.

Grundsätzlich gibt es verschiedene Entscheidungsmöglichkeiten:

- a) Annahme der Konfliktoperation(en)
- 35 b) Ablehnung der Konfliktoperation(en)
- c) Teilweise Annahme, teilweise Ablehnung der Konfliktoperation(en).

d)Ablehnung der Konfliktoption(en), Annahme neuer erzeugter
Operation(en).

Den einzelnen Konflikttypen wird ein Entscheidungsset ES zugeordnet, wobei der Entscheidungsset ES mögliche Entscheidungen enthält, mit denen eine durch eine Operation des jeweiligen Konflikttyps, dem jeweils ein Entscheidungsset ES zugeordnet ist, erzeugte Inkonsistenz behoben werden kann.

10 In der Fig.4 ist eine Zusammenstellung aller Entscheidungssets, die jeweils einem Konflikt zugeordnet sind, dargestellt.

Es wird jeweils in einer Zeile der in Fig.4 dargestellten Tabelle eine mögliche Entscheidungsmöglichkeit E1, E2, E3a, E3b, E4, E5a, E5b, E6 dargestellt.

Mit einem x in einem Feld ist jeweils bezeichnet, daß der jeweils in der Spalte aufgeführte Konflikt durch eine Entscheidungsmöglichkeit, die in der jeweiligen Zeile dargestellt ist, gelöst werden kann.

Im weiteren wird eine Übersicht über die möglichen Entscheidungsmöglichkeiten gegeben:

25 Eine erste Entscheidungsmöglichkeit E1 beschreibt die Annahme einer Konfliktoption oder mehrerer Konfliktoptionen.

Eine Konfliktoption beschreibt alle zu einem Konflikt gehörenden Datenoperationen. Unter Annahme wird verstanden, daß die Konfliktoptionen in der Kopiedatenbank, in der sie noch nicht vorgenommen worden sind, ausgeführt werden.

35 Eine zweite Entscheidungsmöglichkeit E2 beschreibt die Ablehnung einer Konfliktoption oder mehrerer Konfliktoptionen.

Eine dritte Entscheidungsmöglichkeit E3 beschreibt die Annahme einer oder mehrerer Konfliktoperation(en) in einer Kopiedatenbank und die Ablehnung der Konfliktoperation(en) in der anderen Kopiedatenbank.

5

Für die dritte Entscheidungsmöglichkeit E3 ist eine Detailsentscheidung vorgesehen, die definiert, welche der in den Historien verschiedener Kopiedatenbanken der Benutzer A und B vorliegenden Konfliktoperationen angenommen und welche abgelehnt werden sollen.

10

Diese Entscheidungsmöglichkeiten werden als erster Teil E3a der dritten Entscheidungsmöglichkeit E3 und zweiter Teil E3b der dritten Entscheidungsmöglichkeit E3 bezeichnet. Der erste Teil E3a der dritten Entscheidungsmöglichkeit E3 beschreibt die Annahme der Konfliktoperation(en) in der Kopiedatenbank des ersten Benutzers A und die Ablehnung der Konfliktoperation(en) in der Kopiedatenbank des zweiten Benutzers B. Der zweite Teil E3b der dritten Entscheidungsmöglichkeit E3 beschreibt die Annahme der Konfliktoperation(en) der Kopiedatenbank des zweiten Benutzers B und die Ablehnung der Konfliktoperation(en) der Kopiedatenbank des ersten Benutzers A.

15

20

Wie in Fig.4 dargestellt, beschreibt ein erster Entscheidungsset ES1, der dem ersten harmlosen Konflikt HK1 zugeordnet ist, die erste Entscheidungsmöglichkeit E1 sowie die zweite Entscheidungsmöglichkeit E2 zur Erhebung des ersten harmlosen Konflikts HK1.

25

Ein zweiter Entscheidungsset ES2 ist dem zweiten harmlosen Konflikt HK2 zugeordnet und enthält wiederum die erste Entscheidungsmöglichkeit E1 sowie die zweite Entscheidungsmöglichkeit E2 zur Behebung des zweiten harmlosen Konflikts HK2.

30

Entsprechen die bisherigen Lösungsmöglichkeiten durch ein Annehmen oder Ablehnen vorhandener Konfliktoperationen nicht den Zielvorstellungen der Benutzer bezüglich der letztendli-

35

chen reintegrierten Datenbank, d.h. können die Benutzer A und B sich nicht auf eine durch die Konflikte beschriebenen Zustände einigen, so gibt es die Möglichkeit der Annahme einer Zwischenlösung oder die Möglichkeit der Auswahl und Annahme neuer, nicht im Entscheidungsset enthaltener Operationen. Beide Möglichkeiten sollen nachfolgend dargestellt werden.

Für einen Konflikt mit einer Anzahl n den Konflikt definierenden, gleichen Operationen aus einem Set an Datenoperationen in nur einer Historie (Konflikte der Typen HK7 II und HK8 II) gibt es generell im weiteren beschriebene Auswahlmöglichkeiten von Zwischenzuständen.

Für den siebten harmlosen Konflikt HK7 mit $n = 1$ wird nachfolgend HK7 I und für den siebten harmlosen Konflikt HK7 mit $n > 1$ wird nachfolgend HK7 II geschrieben. Für den achten harmlosen Konflikt HK1 mit $m = 1$ wird nachfolgend HK8 I und für den achten harmlosen Konflikt HK8 mit $n > 1$ wird nachfolgend HK8 II geschrieben.

Eine vierte Entscheidungsmöglichkeit E4 beschreibt dabei eine teilweise Annahme und teilweise Ablehnung der Konfliktoperationen.

Für die vierte Entscheidungsmöglichkeit E4 ist eine Präzisierung vorgesehen, die definiert, wieviele der einseitig auftretenden Konfliktoperationen angenommen und wieviele abgelehnt werden sollen. Die Entscheidungsmöglichkeiten reichen bei einer Anzahl von n Operationen (n ist Element der natürlichen Zahlen), von einer Annahme einer Operation und einer Ablehnung von $n-1$ Operationen bis zu einer Annahme von $n-1$ und einer Ablehnung von einer Operation. Als Entscheidungsmöglichkeit reicht dabei aus, die Anzahl k ($0 < k < n$) der angenommenen Operationen zu definieren. Die Anzahl der abgelehnten Operationen errechnet sich dann aus $n - k$. Die vierte Entscheidungsmöglichkeit E4 kann somit folgendermaßen spezia-

lisiert werden: Die vierte Entscheidungsmöglichkeit E4 beschreibt die Anzahl der angenommenen Konfliktoperationen k .

Im weiteren wird der fünfte kritische Konflikt KK5 mit
5 $n = m = 1$ nachfolgend mit KK5 I und für den fünften kritischen Konflikt KK5 mit $n > 1$ und $m > 1$ nachfolgend KK5 II bezeichnet.

Für einen fünften kritischen Konflikt KK5 II mit einer Anzahl
10 n den Konflikt definierenden, gleichen Operationen `changeEigenschaftAdd` bei einer Kopiedatenbank und einer Anzahl m den Konflikt definierenden gleichen Operationen `changeEigenschaftDel` in einer anderen Kopiedatenbank gestaltet sich die Auswahl der Zwischenzustände schwieriger.

15 Da die den Konflikt definierenden Operationen in der einen Historie und die am Konflikt beteiligten Operationen in der anderen Historie einander auslöschend sind, kann das gleiche Endergebnis durch verschiedene Entscheidungsmöglichkeiten ge-
20 troffen werden.

So kann die Rücksetzung einer `changeEigenschaftAdd`-Operation und die Annahme einer `changeEigenschaftAdd`-Operation in der
einen Kopiedatenbank verbunden mit der Rücksetzung einer
25 `changeEigenschaftDel`-Operation bei einer anderen Kopiedatenbank das gleiche Ergebnis erzielen, wie die Annahme zweier `changeEigenschaftAdd`-Operationen der einen Kopiedatenbank und einer `changeEigenschaftDel`-Operation bei der anderen Kopiedatenbank. Damit alle Entscheidungsmöglichkeiten zwischen den
30 Extrema Ablehnung der Operationen der einen Kopiedatenbank und Annahme der Operationen bei der anderen Kopiedatenbank (dritte Entscheidungsmöglichkeit E3a/E3b) gegeben sind, gleichzeitig aber vermieden wird, daß verschiedene Entscheidungsmöglichkeiten ein gleiches Ergebnis liefern, gibt es die
35 unten aufgeführte Lösungsmöglichkeit der fünften Entscheidungsmöglichkeit E5.

Die fünfte Entscheidungsmöglichkeit E5 kann durch Ablehnung aller Operationen den letzten gemeinsamen Stand zwischen den Kopiedatenbanken erzeugen und ermöglicht alle anderen Endzustände aus den Kombinationen der Operationen, jedoch ohne die
5 oben vorliegenden Konfliktzustände.

Die fünfte Entscheidungsmöglichkeit E5 beschreibt also die teilweise Annahme und teilweise Ablehnung der Konfliktoperationen bei einer Kopiedatenbank und die Ablehnung der Konfliktoperationen bei einer anderen Kopiedatenbank.
10

Für die fünfte Entscheidungsmöglichkeit E5 sind die Teilentscheidungsmöglichkeiten notwendig, die einerseits definieren, welche Kopiedatenbank von der teilweisen Annahme und teilweisen Ablehnung und welche Datenbank von der vollständigen Ablehnung betroffen ist und andererseits, wieviele der Konfliktoperationen bei einer teilweisen Annahme angenommen und wieviele abgelehnt werden sollen. Die Definition der Anzahl an Operationen bei teilweiser Annahme und teilweiser Ablehnung
15 kann dabei wie bei der vierten Entscheidungsmöglichkeit E4 über die alleinige Definition der Anzahl der angenommenen Operationen erfolgen. Die Anzahl der angenommenen Operationen in einer ersten Kopiedatenbank des ersten Benutzers A werden dabei mit i und die Anzahl der angenommenen Operationen der
20 Kopiedatenbank des zweiten Benutzers B mit k bezeichnet. Somit sind Detailentscheidungen der fünften Entscheidungsmöglichkeit E5:

Erster Teil E5a der fünften Entscheidungsmöglichkeit E5:
30 Anzahl der angenommenen Konfliktoperationen der Kopiedatenbank des ersten Benutzers A:
($1 < i < n$), wenn es sich um eine changeEigAdd-Operation handelt,
($1 < k < m$), wenn es sich um eine changeEigDel-Operation
35 handelt, und Ablehnung aller Konfliktoperationen der Kopiedatenbank des zweiten Benutzers B.

Zweiter Teil E5b der fünften Entscheidungsmöglichkeit E5:
Anzahl der angenommenen Konfliktoperationen der Kopiedatenbank des zweiten Benutzers B:

- 5 $(1 < i < n)$, wenn es sich um eine changeEigAdd-Operation handelt,
 $(1 < k < m)$, wenn es sich um eine changeEigDel-Operation handelt und Ablehnung aller Konfliktoperationen der Kopiedatenbank des ersten Benutzers A.

- 10 Eine Auswahlmöglichkeit für die Erzeugung eines neuen Zustandes bzgl. des Konfliktes wird durch die nachfolgenden Entscheidungsmöglichkeiten definiert werden.

- 15 Generell ist die Möglichkeit der Erzeugung und Auswahl eines sich von den beiden vorliegenden Versionen unterscheidenden Zustandes für alle Konflikte außer den Konflikten des Typs HK1, HK2, HK4, HK10 gegeben.

- 20 Für eine Erzeugung eines neuen Zustandes bedarf es der Schaffung einer gemeinsamen Ausgangsposition, d.h. die betreffende(n) Operation(en) müssen abgelehnt und beide Kopiedatenbanken bezüglich der von dem Konflikt betroffenen Datenstruktur konsistent gemacht werden. Diese Ablehnung der Operationen ist bei sich überschreibenden Operationen des Typs
25 changeEigenschaft() (HK6, HK4) nicht erforderlich, da die mit dem neuen Zustand erzeugte Operation die alten Operationen direkt überschreibt.

- 30 Für einen Konflikt mit einer den Konflikt definierenden Operation aus dem Set an Datenoperationen (Konflikte der Typen HK3, HK6, HK7 I, HK8 I und HK9), für einen Konflikt mit mehreren den Konflikt definierenden Operationen aus dem Set an Datenoperationen bei nur einer Kopiedatenbank (Konflikte der Typen HK5, HK7 II, HK8 II, HK11) sowie für einen Konflikt mit
35 mindestens einer den Konflikt definierenden Operation aus dem Set an Datenoperationen bei beiden Kopiedatenbanken (Konflikte der Typen KK1, ..., KK8) gibt es folgende Lösungs-

möglichkeit, die als sechste Entscheidungsmöglichkeit E6 bezeichnet wird.

Die sechste Entscheidungsmöglichkeit E6 beschreibt die Ablehnung der Konfliktoperation(en) und Auswahl neuer Operation(en).

Für die beidseitige Änderung einer Beziehung (KK2) oder die beidseitige Änderung eines Eintrages in einer geordneten Aufzählung (KK7) bezieht sich die Ablehnung dabei, im Gegensatz zur zweiten Entscheidungsmöglichkeit E2 (Ablehnung aller Operationen) nur auf die Erzeugeroperationen der neuen Beziehungen bzw. der neuen Einträge. Die gemeinsame Löschoption der alten Beziehung oder des alten Eintrages bleibt unberührt. Für den dritten kritischen Konflikt KK3 und den achten kritischen Konflikt KK8 gibt es bei der sechsten Entscheidungsmöglichkeit E6 nur die Möglichkeit, die createBeziehung-Operation oder die changeEigenschaftInsert-Operation zu ändern. Die gemeinsame Löschoption bleibt unberührt.

Für die Definition eines neuen Zustandes eines Konflikts vom Typ HK7, HK8 oder KK5 werden die Anzahl der changeEigenschaftAdd-Operationen und changeEigenschaftDel-Operationen ebenfalls mit i und k beschrieben (i, wenn es sich um changeEigAdd-Operationen handelt und k, wenn es sich um changeEigDel-Operationen handelt).

Für die Auswahl eines neuen Zustandes und die Erzeugung der dafür notwendigen Operationen sind Interaktionen auf der Oberfläche des Anwendungsprogramms üblich. Besteht die Möglichkeit einer Auswahl eines neuen Zustandes nur über eine komplexe Interaktion und betreffen die durch die Interaktion erzeugten Operationen auch noch andere, nicht gelöste Konflikte, so besteht die Möglichkeit

- a) mit dem Entscheidungsset die anderen betroffenen Konflikte ebenfalls zu lösen oder
b) die Erzeugung des neuen Zustandes auf einen späteren Zeitpunkt, d.h. nach der Reintegration und während der gekoppelten Weiterarbeit zu verschieben.

Sollen nach a) andere betroffene Konflikte ebenfalls gelöst werden, so ist vorher die Ablehnung der Operationen des Konfliktsets notwendig, wenn es sich um Konflikte vom Typ HK6 und KK4 handelt und die Operationen keine sich überschreibenden Operationen sind.

Fig.4 zeigt die Entscheidungssets ES1, ES2, ES3, ES4, ES5, ES6, ES7, ES8, ES9, ES10, ES11, ES12, ES13, ES14, ES15, ES16, ES17, ES18, ES19, ES20, ES21, ES22, die den jeweiligen Konflikten zugeordnet sind.

Dem sechsten harmlosen Konflikt HK6 ist ein sechster Entscheidungsset ES6 zugeordnet, der die erste Entscheidungsmöglichkeit E1, die zweite Entscheidungsmöglichkeit E2 sowie die sechste Entscheidungsmöglichkeit E6 enthält.

Ein zwölftes Entscheidungsset ES12 ist dem ersten kritischen Konflikt KK1 zugeordnet. Der zwölfte Entscheidungsset ES12 enthält vier mögliche Entscheidungen, die zweite Entscheidungsmöglichkeit E2, den ersten Teil E3a der dritten Entscheidungsmöglichkeit E3, den zweiten Teil E3b der dritten Entscheidungsmöglichkeit E3 sowie die sechste Entscheidungsmöglichkeit E6.

Die weiteren Entscheidungssets sind in Fig.4 dargestellt und sollen zur Vereinfachung im folgenden anhand der folgenden Liste dargestellt werden:

- Dem ersten harmlosen Konflikt HK1, dem zweiten harmlosen Konflikt HK2, dem vierten harmlosen Konflikt HK4 sowie dem zehnten harmlosen Konflikt HK10 sind jeweils Entscheidungs-

sets zugeordnet, die die erste Entscheidungsmöglichkeit E1 sowie die zweite Entscheidungsmöglichkeit E2 umfassen.

- Dem dritten harmlosen Konflikt HK3, dem fünften harmlosen Konflikt HK5, dem sechsten harmlosen Konflikt HK6, dem ersten Typ HK7 I des siebten harmlosen Konflikts HK7, dem ersten Typ HK8 I des achten harmlosen Konflikts HK8, dem neunten harmlosen Konflikt HK9 sowie dem elften harmlosen Konflikt HK11 sind jeweils Entscheidungssets zugeordnet, die die erste Entscheidungsmöglichkeit E1, die zweite Entscheidungsmöglichkeit E2 sowie die sechste Entscheidungsmöglichkeit E6 enthalten.
- Dem zweiten Typ HK7 II des siebten harmlosen Konflikts HK7, dem zweiten Typ HK8 II des achten harmlosen Konflikts HK8, dem ersten kritischen Konflikt KK1, dem zweiten kritischen Konflikt KK2, dem dritten kritischen Konflikt KK3, dem vierten kritischen Konflikt KK4, dem ersten Typ KK5 I des fünften kritischen Konflikts KK5, dem sechsten kritischen Konflikt KK6, dem siebten kritischen Konflikt KK7 und dem kritischen Konflikt KK8 sind jeweils Entscheidungssets zugeordnet, die die zweite Entscheidungsmöglichkeit E2, den ersten Teil E3a der dritten Entscheidungsmöglichkeit E3, den zweiten Teil E3b der dritten Entscheidungsmöglichkeit E3 sowie die sechste Entscheidungsmöglichkeit E6 enthalten.
- Dem zweiten Typen KK5 II des fünften kritischen Konflikts KK5 ist ein Entscheidungsset mit sechs möglichen Entscheidungen, der zweiten Entscheidungsmöglichkeit E2, dem ersten Teil E3a der dritten Entscheidungsmöglichkeit E3, dem zweiten Teil E3b der dritten Entscheidungsmöglichkeit E3, dem ersten Teil E5a der fünften Entscheidungsmöglichkeit E5, dem zweiten Teil E5b der fünften Entscheidungsmöglichkeit E5 sowie der sechsten Entscheidungsmöglichkeit E6, zugeordnet.
- Dem zweiten Typ HK7 II des siebten harmlosen Konflikts HK7 und dem zweiten Typ HK8 II des achten harmlosen Konflikts HK8 ist jeweils ein Entscheidungsset mit vier möglichen Entscheidungen, der ersten Entscheidungsmöglichkeit E1, der zweiten Entscheidungsmöglichkeit E2, der vierten Entschei-

dungsmöglichkeit E4 sowie der sechsten Entscheidungsmöglichkeit E6 zugeordnet.

Einschränkungen von Entscheidungsmöglichkeiten

5

Es ist zu bemerken, daß die korrekte Ausführung einzelner Entscheidungen zum Vorhandensein eines Datenelements oder mehrerer Datenelemente in beiden Kopiedatenbanken abhängig ist.

10

Beispielsweise müssen für eine Annahme bei dem dritten harmlosen Konflikt HK3A beide in der Beziehungsoperation in Konflikt über ihre Identifier bezeichneten Datenelemente auch in der Kopiedatenbank des Benutzers B vorhanden sein. Fehlt eines der Datenelemente oder fehlen gar beide, so ist diese Entscheidung über eine Annahme der Operation nicht möglich.

15

Somit ist erkennbar, daß zwischen einzelnen Konflikten Abhängigkeiten bestehen können.

20

Eine Abhängigkeit eines Konflikts zu Konflikten des Typs HK1A, HK1B, HK2A oder HK2B bezüglich seiner Entscheidungsmöglichkeiten ist wie folgt definiert:

25 Abhängiger Konflikt:

Ein Konflikt ist von dem ersten harmlosen Konflikt HK1 oder dem zweiten harmlosen Konflikt HK2 dann abhängig, wenn seine Entscheidungsmöglichkeiten durch das Vorhandensein eines ersten harmlosen Konflikts HK1 oder eines zweiten harmlosen

30

Konflikts HK2 eingeschränkt werden.

Eine Abhängigkeit AK eines Konflikts ist wie folgt definiert:

Abhängigkeit AK eines Konflikts:

Eine Abhängigkeit AK eines Konfliktes $K1(EHA, EHB, GH)$ von einem Konflikt $K1(EHA, EHB, GH)$ ist ein 4-Tupel

5 AK ($K1, K2$) def ($id, ktyp, kid1, kid2$)

- id ist ein systemweit eindeutiger Identifier (siehe auch Definition eines Datenelementes)
- $ktyp \in \{HK1A, HK1B, HK2A, HK2B\}$
- 10 • $kid1 = K1.id$
- $kid2 = K2.id$.

Ein Abhängigkeitsregister AKR ist wie folgt definiert:

15 **Abhängigkeitsregister AKR:**

Ein Abhängigkeitsregister AKR zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 1-Tupel

AKR (EH_A, EH_B, GH) def (abhängigkeitsids)

20

- abhängigkeitsids sind die Identifier aller erkannten Abhängigkeiten der Historien EH_A und EH_B und der gemeinsamen Historie GH .

25 Alle Einschränkungen von Entscheidungsmöglichkeiten durch vorhandene Konflikte des ersten harmlosen Konflikts $HK1$ und des zweiten harmlosen Konflikts $HK2$ werden zu Beginn der Konfliktlösung auf der Grundlage der Abhängigkeiten AK erkannt und markiert.

30

Dazu dient der in der Konfliktdefinition für jeden Konflikt eingeführte Parameter der Entscheidungseinschränkungen. Nachfolgend werden alle möglichen Einschränkungen beschrieben.

35 Der erste harmlose Konflikt $HK1$ beeinträchtigt die Entscheidungsmöglichkeiten von Konflikten mit Operationen innerhalb der eigenen Historie. Zu diesen abhängigen Konflikten gehören

alle, die eine createBeziehung-Operation oder eine Eigenschaftsänderung mit dem erzeugten Datenelement enthalten. Dabei werden bei harmlosen Konflikten HK6, ..., HK11 mit Eigenschaftsoperationen zu diesem Datenelement die Entscheidungen um eine Annahme und eine Erzeugung eines neuen Zustandes minimiert.

Die Entscheidungsmöglichkeiten der kritischen Konflikte KK1, KK2 und KK3 mit Beziehungsoperationen mit dem erzeugten Datenelement werden um die Möglichkeit der Annahme der Operationen (-E3a, -E3b) der Kopiedatenbank reduziert, in dessen Historie die createElement-Operation steht. Die Entscheidungsmöglichkeiten für den dritten harmlosen Konflikt HK3 und den fünften harmlosen Konflikt HK5 mit Beziehungsoperationen mit dem erzeugten Datenelement werden um die Entscheidung der Übernahme und/oder der Erzeugung eines neuen Zustandes reduziert. Kritische Konflikte mit Eigenschaftsoperationen erfahren keine Veränderungen ihrer Entscheidungsmenge (Entscheidungssets).

Ein zweiter harmloser Konflikt HK2 beeinträchtigt die Entscheidungsmöglichkeit von Konflikten beider Kopiedatenbanken. Die harmlosen Konflikte zur Veränderung von Eigenschaften HK6, ..., HK11 werden wie bei dem ersten harmlosen Konflikt HK1 behandelt. Die harmlosen Konflikte mit Beziehungsoperationen (HK4, HK5) in der Historie der deleteElement-Operation besitzen keine Entscheidungsmöglichkeit zur Ablehnung mehr und die Entscheidungen zur harmlosen Beziehungskonflikten der anderen Kopiedatenbank HK3, HK5 werden um die Möglichkeit der Übernahme und/oder Erzeugung eines neuen Zustandes minimiert. Die kritischen Konflikte mit Beziehungsoperationen der Kopiedatenbank ohne die deleteElement-Operation werden um die Möglichkeit des Rücksetzens und/oder der Annahme der Operationen reduziert. Kritische Konflikte mit Eigenschaftsoperationen erfahren auch hier keine Veränderung ihrer Entscheidungsmenge.

Eine gemeinsame deleteElement-Operation, die in der gemeinsamen Historie GH enthalten ist und keinen Konflikt darstellt, verringert in speziellen Fällen die Entscheidungsmöglichkeiten. So können alle kritischen Konflikte KK2 und KK3 mit Beziehungsoptionen, in denen das Zieldatenelement der gemeinsamen deleteBeziehung-Operation dem beidseitig gelöschten Datenelement entspricht, nicht mehr rückgesetzt werden.

Abhängige Konflikte mit einer oder mehreren Beziehungsoptionen (HK3, HK4, HK5, KK1, KK2, KK3), d.h. mit mehreren verzeichneten Identifiern und damit mehreren beteiligten Datenelementen, können gleichzeitig mehrere Abhängigkeiten aufweisen. Dabei kann es pro auftretenden Identifier in einem Konflikt nur eine Abhängigkeit geben. Ein abhängiger Konflikt mit mehreren Identifiern kann einerseits mehrere Abhängigkeiten zu Konflikten desselben Konflikttyps (zum Beispiel zu zwei Konflikten des ersten harmlosen Konflikts HK1a) und andererseits mehrerer Abhängigkeiten zu Konflikten unterschiedlicher Konflikttypen (z.B. zu einem Konflikt vom Typ HK1a und zu einem Konflikt vom Typ HK1b) aufweisen. Beispielsweise kann ein abhängiger Konflikt vom Typ HK3a zwei Abhängigkeiten zu zwei Konflikten des Typs HK1a haben, nämlich eine mit dem Identifier id1 und eine mit dem Identifier id2. Der zweite kritische Konflikt KK2 kann gleichzeitig eine Abhängigkeit zu dem zweiten harmlosen Konflikt HK2a, zu dem ersten harmlosen Konflikt HK1a und zu dem ersten harmlosen Konflikt HK1b aufweisen.

Durch die Abhängigkeit der Identifier zu einem Konflikt vom Typ HK1 oder HK2 kann es maximal pro Konflikt vom Typ HK5, KK1, KK2 oder KK3 drei Einschränkungen geben. Alle Konflikte vom Typ HK3A, HK3B, HK4A, HK4B, HK5A und HK5B können gleichzeitig mehrere Abhängigkeiten zu Konflikten desselben oder unterschiedlichen Typs haben. Konflikte der Typen KK1, KK2, KK3A und KK3B können dagegen gleichzeitig mehrere Abhängigkeiten zu Konflikten verschiedener Typen aufweisen. Es können jedoch bei mehrfach vom selben Konflikttyp abhängigen Kon-

flikten auch die gleichen Einschränkungen der Entscheidungsmöglichkeiten mehrfach auftreten. So sind für einen Konflikt vom Typ HK3A gleichzeitig zwei Abhängigkeiten bei Vorhandensein eines Konflikts vom Typ HK1A oder eines vom Typ HK2B möglich und die Entscheidungen werden jeweils um die erste Entscheidung E1 minimiert.

Die sich zu manchen abhängigen Konflikten ergebenden mehrfachen Einschränkungen mit den gleichen Entscheidungsmöglichkeiten bedürfen keiner gesonderten Betrachtung. Jede dieser Entscheidungseinschränkungen wird betrachtet, als ob es eine eigene, spezielle Einschränkung ist. Es werden somit alle mehrfachen Einschränkungen, wie andere auch, im Konflikt vermerkt. Bei einer Lösung eines Konflikts des Typs HK1 und HK2 in der Art, daß eine der mehrfach vorkommenden Einschränkungen aufgehoben wird, bleiben die restlichen dieser Einschränkungen erhalten. Erst wenn durch verschiedene Konfliktlösungen keine der mehrfachen Einschränkungen mehr vorhanden ist, kann eine Entscheidung dieser Art vorgenommen werden. Dies gilt unabhängig davon, ob die Einschränkung einmal oder mehrmals vorhanden war.

Die einmalig zu Beginn der Reintegration erkannten Einschränkungen durch Abhängigkeiten der Konflikte zu Konflikten des Typs HK1 und HK2 werden abhängig von der Lösung der Konflikte des Typs HK1 und HK2 während der Reintegration dynamisch geändert. So kommt es abhängig vom Typ der abhängigen Konflikte und der jeweiligen Lösungsentscheidung der createElement-Operation und deleteElement-Operation zu den nachfolgenden

Änderungen der Entscheidungseinschränkungen:

- a) Die Annahme einer createElement-Operation (erste Entscheidung E1 zu einem Konflikt vom Typ HK1) verursacht bei allen von der Operation abhängigen Konflikten eine Rücksetzung der Entscheidungseinschränkungen, d.h. eine Erweiterung der Entscheidungsmöglichkeiten.
- b) Die Ablehnung einer deleteElement-Operation (zweite Entscheidung E2 zu einem Konflikt vom Typ HK2) verursacht

ebenfalls eine Rücksetzung der Entscheidungseinschränkungen, d.h. eine Erweiterung der Entscheidungsmöglichkeiten für die von diesem Konflikt abhängigen Konflikte.

- 5 c) Die Ablehnung einer createElement-Operation (zweite Entscheidung E2 zu einem Konflikt vom Typ HK1) führt zur Beibehaltung der bereits vorgenommenen Entscheidungseinschränkungen der von diesem Konflikt abhängigen Konflikte. Es ändern sich somit keine Entscheidungsmöglichkeiten.
- 10 d) Die Annahme einer deleteElement-Operation (erste Entscheidung E1 zu einem Konflikt vom Typ HK2) führt zur Beibehaltung der bereits vorgenommenen Entscheidungseinschränkungen der von diesem Konflikt abhängigen Konflikte. Es ändern sich somit keine Entscheidungsmöglichkeiten.

- 15 Eine einseitig erzeugte Anomalie kann über jeweils zwei vorhandene Konflikte in einer Historie erkannt werden. Dabei gibt es die Möglichkeiten der Konfliktpaare HK5Aa/HK5Ab, HK4A/HK3A, HK4A/HK5A. Bei einem Konflikt vom Typ HK5Aa ist die deleteBeziehung-Operation und bei einem Konflikt vom Typ
- 20 HK5Ab ist die createBeziehung-Operation an der Anomalie beteiligt.

- Für Anomalien, die durch Veränderungen auf der Kopiedatenbank des zweiten Benutzers B entstehen, gelten die gleichen Möglichkeiten: HK5Ba/HK5Bb, HK4B/HK3B, HK4B/HK5B. Allen Konfliktpaaren ist dabei gemeinsam, daß es eine deleteBeziehung-Operation mit einem Identifier id1 als Quelldatenelement und einem Identifier id2 als Zieldatenelement in einem der beiden Konflikte gibt und die gleichen Identifier vertauscht als
- 25 Quelldatenelement und Zieldatenelement in einer createBeziehung-Operation des anderen Konfliktes auftreten.
- 30

- Die Annahme des Konflikts der einseitigen Anomalie, wie oben beschrieben, mit der createBeziehung-Operation verhindert die
- 35 Ablehnung des Konflikts der einseitigen Anomalie mit der deleteBeziehung-Operation einer Ablehnung des Konflikts der einseitigen Anomalie mit der deleteBeziehung-Operation dage-

gen verhindert die Annahme des Konflikts der Anomalie mit der createBeziehung-Operation. Die Änderung einer der beiden Konflikte vermindert die Entscheidungsmöglichkeiten des anderen Konfliktes nicht.

5

Somit ergibt sich, daß eine einseitig erzeugte Anomalie gerichteter Beziehungen durch eine Ablehnung beider Konflikte, einer Annahme beider Konflikte, einer verschiedenen Änderung beider Konflikte oder einer Veränderung einer der Konflikte und einer Annahme oder Ablehnung des anderen Konflikts lösbar ist.

10

Eine beidseitig erzeugte Anomalie gerichteter Beziehungen ist durch die Entscheidung einer der beiden Konflikte und eine davon verschiedene Entscheidung des anderen Konfliktes lösbar.

15

Zur Vermeidung der Entstehung einer Anomalie (sogenannte Pseudo-Anomalie), wie oben beschrieben, sind folgende Einschränkungen der Entscheidungsmöglichkeiten vorgesehen:

20

a) Nach einer Annahme eines Konflikts mit der createBeziehung-Operation, die das gemeinsame Datenelement id1 als Zieldatenelement enthält (createBeziehung21) darf es für den Konflikt mit den zwei createBeziehung-Operationen, die das gemeinsame Datenelement als Quelldatenelement enthalten, keine Entscheidungsmöglichkeit der sechsten Entscheidungsmöglichkeit E6 mit einer Ersetzung des Zieldatenelements (idx oder idz) durch das Quelldatenelement id2(createBeziehung21) mehr geben.

25

b) Nach einer erfolgten sechsten Entscheidung E6 aufgrund der sechsten Entscheidungsmöglichkeit E6 und der Auswahl eines neuen Zieldatenelements id2 für den Konflikt mit den zwei createBeziehung-Operationen, die das gemeinsame Datenelement id1 als Quelldatenelement enthalten, darf keine Annahme des Konfliktes mit der createBeziehung-Operation mit dem gemeinsamen Datenelement id1 als Zieldatenelement (createBeziehung21) mehr möglich sein.

30
35

Wie oben beschrieben, werden im Rahmen dieses Verfahrens die Entscheidungsmöglichkeiten von Konflikten entsprechend deren Abhängigkeiten, Anomalien und Pseudo-Anomalien eingeschränkt.

5

Nach jeder Entscheidung zu einem Konflikt wird eine Änderung der Entscheidungsmöglichkeiten der von dem eben gelösten Konflikt abhängigen oder sich in einer gemeinsamen Anomalie bzw. Pseudo-Anomalie mit diesem Konflikt befindlichen Konflikte

10 entsprechend den Abhängigkeiten, Anomalien und Pseudo-Anomalien vorgenommen.

Für jeden Konflikt wird, wie oben beschrieben, eine Entscheidung getroffen. Die Entscheidung kann auf unterschiedliche

15 Weise erfolgen. Eine Übersicht über mögliche Entscheidungsvariationen ist in [1] zu finden.

Im Rahmen dieses Ausführungsbeispiels ist vorgesehen, daß eine Datenbank oder Kopiedatenbank als Referenzdatenbank angesehen wird, und der Abgleich gemäß der Referenzdatenbank erfolgt.

20

Es wird also, wie in Fig.1 durch eine rekursive Schleife über einem Überprüfungsschritt (Schritt 105) dargestellt ist,

25 überprüft, ob noch ein Konflikt vorliegt und somit eine Entscheidung getroffen werden muß. Ist eine Entscheidung noch zu treffen, so wird diese getroffen. Sind keine Konflikte mehr vorhanden, so wird ein letzter Verfahrensschritt

30 (Schritt 106) durchgeführt, das Abspeichern der reintegrierten Datenbank, welche keine Inkonsistenzen mehr aufweist.

Die konsistenzfreie Datenbank wird wieder an alle weiteren Rechner, die mit dem ersten Rechner 200 verbunden sind, übertragen (Schritt 107).

35

Damit besitzen alle Rechner eine konsistente Kopiedatenbank.

Im weiteren werden einige Alternativen zu dem oben beschriebenen Ausführungsbeispiel dargestellt:

5 Die Erkennung von Inkonsistenzen kann auch nach einer vorgebbaren Anzahl erfolgter Behebung einer Inkonsistenz durch die Suche nach einer weiteren Inkonsistenz erfolgen. Dies kann dahingehend erweitert werden, daß erneut nach jeder Behebung einer Inkonsistenz die Suche nach einer nächsten Inkonsistenz und deren Behebung erfolgt.

10

Es ist ferner möglich, daß durch den ersten Rechner gemäß dem oben dargestellten Verfahren eine Folge von Korrekturbefehlen (Korrektursequenzen) ermittelt wird, die jeweils dem Rechner, dessen Kopiedatenbank auf Inkonsistenzen hin überprüft wurde, 15 übermittelt wird und der jeweilige Rechner anhand der Korrektursequenz seine Kopiedatenbank an die Datenbank abgleicht.

Es ist ferner in einer alternativen Ausführungsform ebenfalls möglich, einem Benutzer oder mehreren Benutzern die Entscheidung zu überlassen, d.h. die Entscheidungsmöglichkeiten werden einem Benutzer auf dem Bildschirm dargestellt, und der 20 Benutzer wählt über die Tastatur oder die Computermaus die von ihm gewünschte Entscheidung aus, die dann von dem Rechner durchgeführt wird.

In diesem Dokument ist folgende Veröffentlichung zitiert:

[1] DE 196 07 132 A1

Patentansprüche

1. Verfahren zur rechnergestützten Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht,
 - a) bei dem mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, definierten Konflikttypen zugeordnet ist,
 - b) bei dem jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
 - c) bei dem die Inkonsistenz unter Verwendung des Entscheidungssets behoben wird.
2. Verfahren nach Anspruch 1, bei dem mehrere Inkonsistenzen behoben werden.
3. Verfahren nach Anspruch 1 oder 2, bei dem jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mehrere Operationen des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
4. Verfahren nach einem der Ansprüche 1 bis 3, bei dem die Datenbankmenge mehrere Kopiedatenbanken der Datenbank aufweist.
5. Verfahren nach einem der Ansprüche 1 bis 4, bei dem vor der Behebung der Inkonsistenzen alle Inkonsistenzen und deren Abhängigkeiten voneinander, Anomalien und Pseudo-Anomalien ermittelt werden.
6. Verfahren nach einem der Ansprüche 1 bis 5,

bei dem das Entscheidungsset eines Konfliktes während des Verfahrens verändert wird.

7. Verfahren nach einem der Ansprüche 1 bis 6,
5 bei dem die Änderung des jeweiligen Entscheidungssets abhängig von Abhängigkeiten von Inkonsistenzen erfolgt.

8. Verfahren nach einem der Ansprüche 1 bis 7,
bei dem nach einer vorgebbaren Anzahl von behobenen Inkonsistenzen die Datenbankmenge auf weitere Inkonsistenzen und deren Abhängigkeiten, Anomalien und Pseudo-Anomalien hin untersucht wird.
10

9. Verfahren nach einem der Ansprüche 1 bis 8,
15 bei dem die Datenbankmenge eine objektorientierte Datenbank enthält.

10. Verfahren nach einem der Ansprüche 1 bis 9,
eingesetzt im Rahmen der objektorientierten Softwareentwicklung.
20

11. Verfahren nach einem der Ansprüche 1 bis 9,
eingesetzt im Rahmen der Erstellung eines strukturierten elektronischen Dokuments.
25

12. Anordnung zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht,
30

mit mindestens einem Prozessor, der derart eingerichtet ist, daß folgende Schritte durchführbar sind:

- a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
35
b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit de-

nen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.

5

13. Anordnung nach Anspruch 12,
bei der der Prozessor derart eingerichtet ist, daß mehrere Inkonsistenzen behoben werden.

10 14. Anordnung nach Anspruch 12 oder 13,
bei der der Prozessor derart eingerichtet ist, daß jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mehrere Operationen des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,

15

15. Anordnung nach einem der Ansprüche 12 bis 14,
bei der der Prozessor derart eingerichtet ist, daß die Datenbankmenge mehrere Kopiedatenbanken der Datenbank aufweist.

20

16. Anordnung nach einem der Ansprüche 12 oder 15,
bei der der Prozessor derart eingerichtet ist, daß vor der Behebung der Inkonsistenzen alle Inkonsistenzen und deren Abhängigkeiten voneinander, Anomalien und Pseudo-Anomalien ermittelt werden.

25

17. Anordnung nach einem der Ansprüche 12 bis 16,
bei der der Prozessor derart eingerichtet ist, daß das Entscheidungsset eines Konfliktes während des Verfahrens verändert wird.

30

18. Anordnung nach einem der Ansprüche 12 bis 17,
bei der der Prozessor derart eingerichtet ist, daß die Änderung des jeweiligen Entscheidungssets abhängig von Abhängigkeiten von Inkonsistenzen erfolgt.

35

19. Anordnung nach einem der Ansprüche 12 bis 18,

bei der der Prozessor derart eingerichtet ist, daß nach einer vorgebbaren Anzahl von behobenen Inkonsistenzen die Datenbankmenge auf weitere Inkonsistenzen und deren Abhängigkeiten, Anomalien und Pseudo-Anomalien hin untersucht wird.

5

20. Anordnung nach einem der Ansprüche 12 bis 19, bei der der Prozessor derart eingerichtet ist, daß die Datenbankmenge eine objektorientierte Datenbank enthält.

10 21. Anordnung nach einem der Ansprüche 12 bis 20, eingesetzt im Rahmen der objektorientierten Softwareentwicklung.

22. Anordnung nach einem der Ansprüche 12 bis 21,
15 eingesetzt im Rahmen der Erstellung eines strukturierten elektronischen Dokuments.

23. Satz mehrerer Anordnungen zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie
20 wie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht,
bei dem jede Anordnung mindestens einen Prozessor aufweist, der derart eingerichtet ist, daß folgende Schritte durchführ-
25 bar sind:

- a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
 - b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet,
30 mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
 - c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben, und
- 35 bei dem die Anordnungen miteinander koppelbar sind.

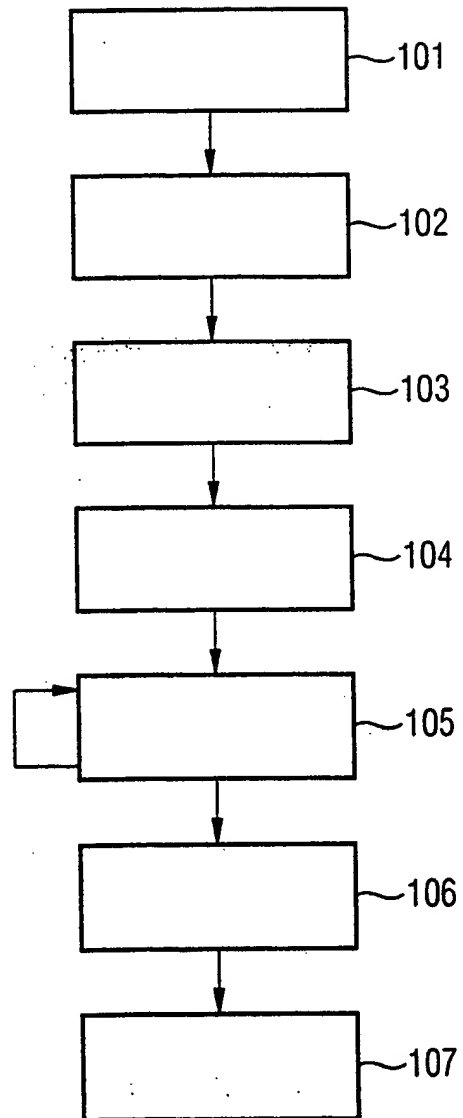
Zusammenfassung

Verfahren, Anordnung und Satz mehrerer Anordnungen zur Behebung von Inkonsistenzen in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist

Zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Daten in der Datenbank oder in der Kopiedatenbank entsteht, ist mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, definierten Konflikttypen zugeordnet. Jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann. Die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben. Eine fehlerfreie Behebung von Inkonsistenzen wird durch Anpassung des Entscheidungssets an die jeweilige Situation gewährleistet.

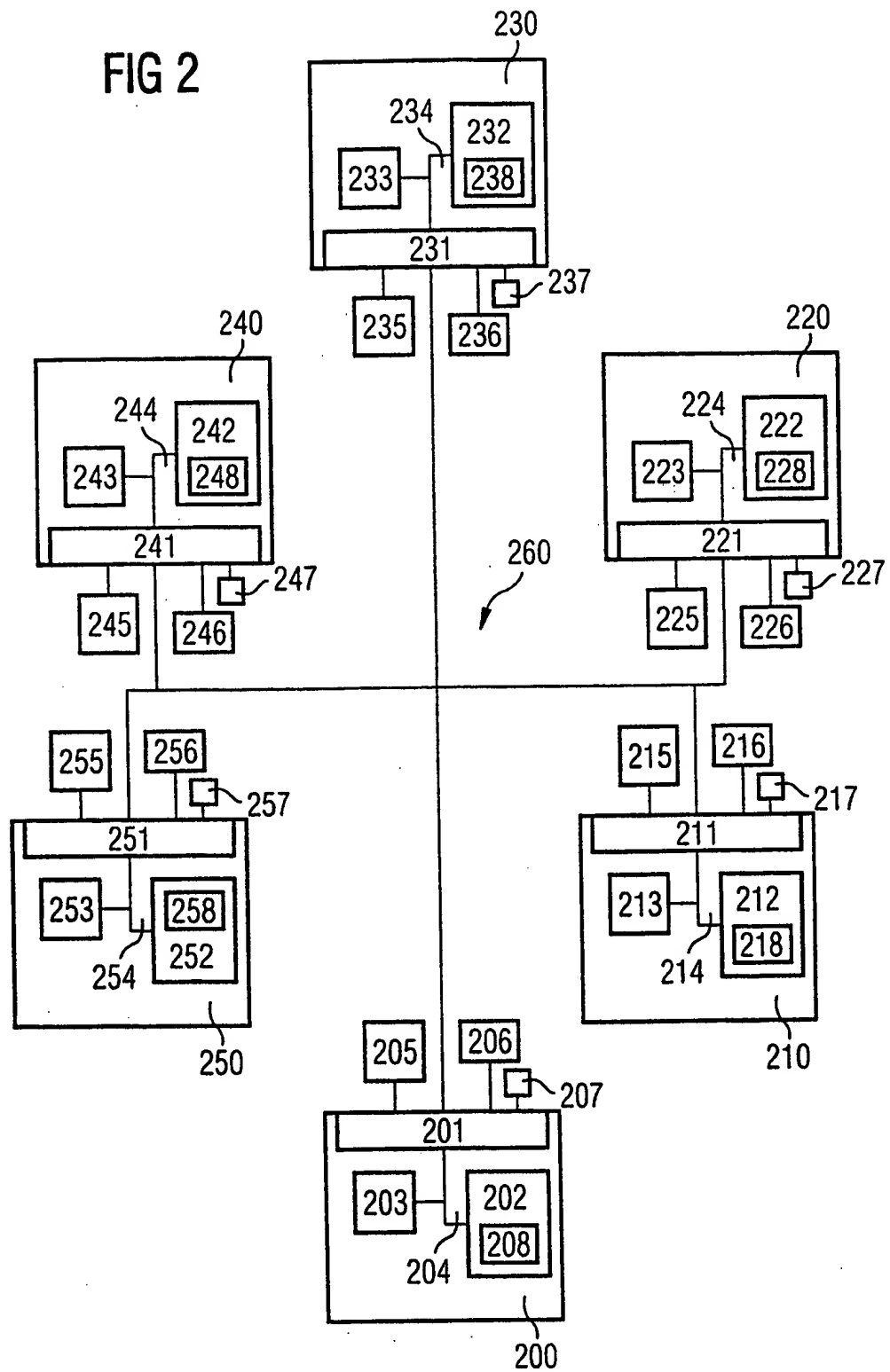
1/4

FIG 1



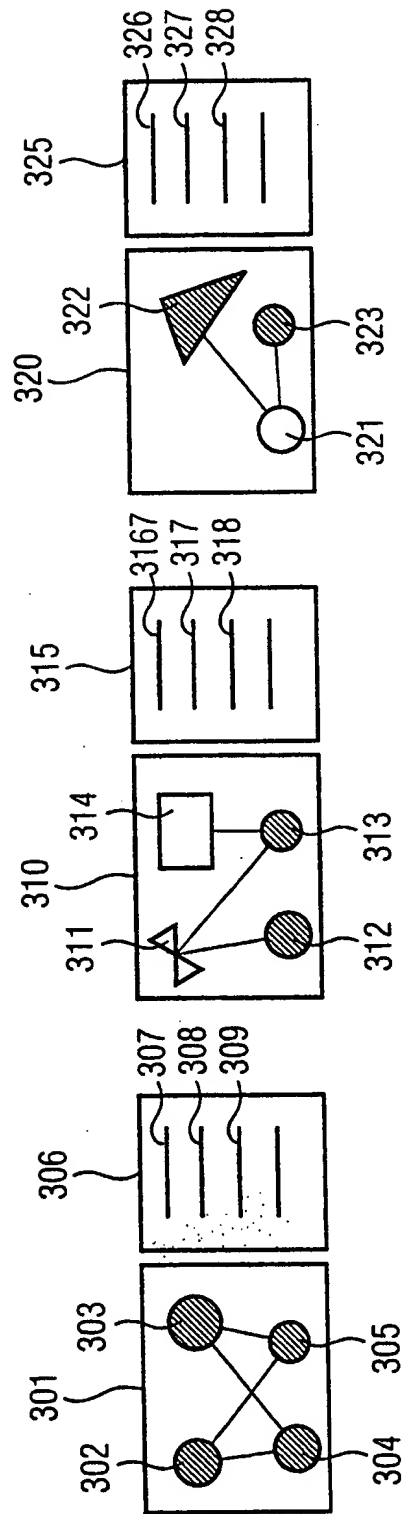
2/4

FIG 2



3/4

FIG 3



INTERNATIONAL SEARCH REPORT

Internal J Application No

PCT/DE 99/01999

A. CLASSIFICATION OF SUBJECT MATTER

IPC 7 G06F17/30

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	DATE C J: "DISTRIBUTED DATABASES" , US, READING, ADDISON WESLEY, PAGE(S) 291-340-309 XP002016220 page 318, line 7 -page 323, line 15 ---	1-6, 12-17, 23
X	ROTHNIE J B ET AL: "INTRODUCTION TO A SYSTEM FOR DISTRIBUTED DATABASES (SDD-1)" ACM TRANSACTIONS ON DATABASE SYSTEMS, US, ASSOCIATION FOR COMPUTING MACHINERY, NEW YORK, vol. 5, no. 1, page 1-17 XP000747926 ISSN: 0362-5915 page 7, line 33 -page 10, line 15 ---	1-6, 12-17, 23
A	EP 0 716 385 A (XEROX CORP) 12 June 1996 (1996-06-12) column 5, line 58 -column 9, line 11 ---	1-8, 12-19, 23
	--- -/--	

☒ Further documents are listed in the continuation of box C.☒ Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search

11 January 2000

Date of mailing of the international search report

17/01/2000

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentplan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Authorized officer

Abbing, R

INTERNATIONAL SEARCH REPORT

Information on patent family members

Internat Application No

PCT/DE 99/01999

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 0716385 A	12-06-1996	US 5671407 A	23-09-1997
		JP 2731373 B	25-03-1998
		JP 8241235 A	17-09-1996
<hr/>			

A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES
IPK 7 G06F17/30

Nach der Internationalen Patentklassifikation (IPK) oder nach der nationalen Klassifikation und der IPK

B. RECHERCHIERTE GEBIETE

Recherchierte Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)
IPK 7 G06F

Recherchierte aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
X	DATE C J: "DISTRIBUTED DATABASES", US, READING, ADDISON WESLEY, PAGE(S) 291-340-309 XP002016220 Seite 318, Zeile 7 -Seite 323, Zeile 15 ---	1-6, 12-17, 23
X	ROTHNIE J B ET AL: "INTRODUCTION TO A SYSTEM FOR DISTRIBUTED DATABASES (SDD-1)" ACM TRANSACTIONS ON DATABASE SYSTEMS, US, ASSOCIATION FOR COMPUTING MACHINERY, NEW YORK, Bd. 5, Nr. 1, Seite 1-17 XP000747926 ISSN: 0362-5915 Seite 7, Zeile 33 -Seite 10, Zeile 15 ---	1-6, 12-17, 23
A	EP 0 716 385 A (XEROX CORP) 12. Juni 1996 (1996-06-12) Spalte 5, Zeile 58 -Spalte 9, Zeile 11 ---	1-8, 12-19, 23
	--- -/--	

☒ Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen☒ Siehe Anhang Patentfamilie

* Besondere Kategorien von angegebenen Veröffentlichungen :

"A" Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist

"E" älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist

"L" Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)

"O" Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht

"P" Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist

"T" Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist

"X" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfindertätiger Tätigkeit beruhend betrachtet werden

"Y" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erfindertätiger Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann nahelegend ist

"Z" Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche

11. Januar 2000

Absenddatum des internationalen Recherchenberichts

17/01/2000

Name und Postanschrift der internationalen Recherchenbehörde

Europäisches Patentamt, P.B. 5818 Patentlaan 2
NL - 2260 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Bevollmächtigter Bediensteter

Abbing, R

INTERNATIONALER RECHERCHENBERICHT

Angaben zu Veröffentlichungen, die zur selben Patentfamilie gehören

Internat. Aktenzeichen

PCT/DE 99/01999

Im Recherchenbericht angeführtes Patentdokument	Datum der Veröffentlichung	Mitglied(er) der Patentfamilie	Datum der Veröffentlichung
EP 0716385 A	12-06-1996	US 5671407 A	23-09-1997
		JP 2731373 B	25-03-1998
		JP 8241235 A	17-09-1996
<hr/>			

C.(Fortsetzung) ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
A	MUTH P ET AL: "SEMANTIC CONCURRENCY CONTROL IN OBJECT-ORIENTED DATABASE SYSTEMS" PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON DATA ENGINEERING,US,LOS ALAMITOS, IEEE COMP. SOC. PRESS, Bd. CONF. 9, Seite 233-242 XP000380184 ISBN: 0-8186-3570-3 das ganze Dokument -----	1,9,12, 20

INTERNATIONAL SEARCH REPORT

Internati	Application No
PCT/DE 99/01999	

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>MUTH P ET AL: "SEMANTIC CONCURRENCY CONTROL IN OBJECT-ORIENTED DATABASE SYSTEMS" PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON DATA ENGINEERING, US, LOS ALAMITOS, IEEE COMP. SOC. PRESS, vol. CONF. 9, page 233-242 XP000380184 ISBN: 0-8186-3570-3 the whole document -----</p>	<p>1, 9, 12, 20</p>

09-1744225
Translation

2

PATENT COOPERATION TREATY

PCT

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

(PCT Article 36 and Rule 70)

Applicant's or agent's file reference GR 98P2167P	FOR FURTHER ACTION See Notification of Transmittal of International Preliminary Examination Report (Form PCT/IPEA/416)	
International application No. PCT/DE99/01999	International filing date (day/month/year) 01 July 1999 (01.07.99)	Priority date (day/month/year) 27 July 1998 (27.07.98)
International Patent Classification (IPC) or national classification and IPC G06F 17/30		
Applicant SIEMENS AKTIENGESELLSCHAFT		RECEIVED APR 19 2001 Technology Center 2100

1. This international preliminary examination report has been prepared by this International Preliminary Examining Authority and is transmitted to the applicant according to Article 36.

2. This REPORT consists of a total of 4 sheets, including this cover sheet.

☒ This report is also accompanied by ANNEXES, i.e., sheets of the description, claims and/or drawings which have been amended and are the basis for this report and/or sheets containing rectifications made before this Authority (see Rule 70.16 and Section 607 of the Administrative Instructions under the PCT).

These annexes consist of a total of 4 sheets.

3. This report contains indications relating to the following items:

- I ☒ Basis of the report
- II ☐ Priority
- III ☐ Non-establishment of opinion with regard to novelty, inventive step and industrial applicability
- IV ☐ Lack of unity of invention
- V ☒ Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement
- VI ☐ Certain documents cited
- VII ☐ Certain defects in the international application
- VIII ☐ Certain observations on the international application

Date of submission of the demand 18 February 2000 (18.02.00)	Date of completion of this report 07 November 2000 (07.11.2000)
Name and mailing address of the IPEA/EP	Authorized officer
Facsimile No.	Telephone No.

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No.

PCT/DE99/01999

I. Basis of the report

1. This report has been drawn on the basis of *(Replacement sheets which have been furnished to the receiving Office in response to an invitation under Article 14 are referred to in this report as "originally filed" and are not annexed to the report since they do not contain amendments.)*:

- ☐ the international application as originally filed.
- ☒ the description, pages 1-52, as originally filed,
 pages _____, filed with the demand,
 pages _____, filed with the letter of _____,
 pages _____, filed with the letter of _____.
- ☒ the claims, Nos. _____, as originally filed,
 Nos. _____, as amended under Article 19,
 Nos. _____, filed with the demand,
 Nos. 1-23, filed with the letter of 23 October 2000 (23.10.2000),
 Nos. _____, filed with the letter of _____.
- ☒ the drawings, sheets/fig 1/4-4/4, as originally filed,
 sheets/fig _____, filed with the demand,
 sheets/fig _____, filed with the letter of _____,
 sheets/fig _____, filed with the letter of _____.

2. The amendments have resulted in the cancellation of:

- ☐ the description, pages _____
- ☐ the claims, Nos. _____
- ☐ the drawings, sheets/fig _____

3. ☐ This report has been established as if (some of) the amendments had not been made, since they have been considered to go beyond the disclosure as filed, as indicated in the Supplemental Box (Rule 70.2(c)).

4. Additional observations, if necessary:

INTERNATIONAL PRELIMINARY EXAMINATION REPORT

International application No.
PCT/DE 99/01999

V. Reasoned statement under Article 35(2) with regard to novelty, inventive step or industrial applicability; citations and explanations supporting such statement

1. Statement

Novelty (N)	Claims	1-23	YES
	Claims		NO
Inventive step (IS)	Claims	1-23	YES
	Claims		NO
Industrial applicability (IA)	Claims	1-23	YES
	Claims		NO

2. Citations and explanations

Reference is made to document D1, DE-A1-196 07 132, cited in the application.

Independent Claim 1 meets the requirements of PCT Article 33(3).

Document D1 is the closest prior art. It describes a method for resolving inconsistencies in a group of databases having a database and at least one copy database of the database, in which at least two modification operations are combined to form a series of operations which are collectively compared.

The method defined in Claim 1 differs from that described in D1 in the following points: (a) at least some of the operations which produce an inconsistency are assigned to defined conflict types; (b) each conflict type is assigned to a decision set which indicates possible decisions by means of which an inconsistency produced by at least one operation of the conflict type in question can be resolved; (c) the inconsistency is resolved using the decision set.

Technical problem: To indicate a further method for

resolving inconsistencies in a group of databases.

Inventive step: The invention enables an inconsistency in a complex database to be resolved generically. As a result of combining possible decisions by means of which a specific problem (an inconsistency) can be resolved and by assigning decision sets to pre-defined conflict types, the invention is highly flexible (many types of action are possible). In one embodiment of the invention, these decision sets can be modified (restricted) as a function of what the inconsistencies are dependent on, which saves further computational time during error resolution. The prior art makes no suggestion of a method having any of the solution features (a), (b) or (c).

Dependent Claims 12 and 23 respectively define the corresponding configuration and set of a plurality of configurations and also meet the requirements of PCT Article 33(3).

The dependent claims also meet the requirements of PCT Article 33(3).

VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS

Absender: INTERNATIONALE RECHERCHENBEHÖRDE

PCT

An
SIEMENS AG
Postfach 22 16 34
D-80506 München
GERMANY

MITTEILUNG ÜBER DIE ÜBERMITTLUNG DES
INTERNATIONALEN RECHERCHENBERICHTS
ODER DER ERKLÄRUNG

(Regel 44.1 PCT)

ZT GG VM Mch P/Ri

Eing. 19. Jan. 2000

GR
Frist

Absendedatum
(Tag/Monat/Jahr)

17/01/2000

Aktenzeichen des Anmelders oder Anwalts

GR 98P2167P

WEITERES VORGEHEN

siehe Punkte 1 und 4 unten

Internationales Aktenzeichen

PCT/DE 99/ 01999

Internationales Anmeldedatum

(Tag/Monat/Jahr)

01/07/1999

Anmelder

SIEMENS AKTIENGESELLSCHAFT et al.

1. ☒ Dem Anmelder wird mitgeteilt, daß der internationale Recherchenbericht erstellt wurde und ihm hiermit übermittelt wird.

Einreichung von Änderungen und einer Erklärung nach Artikel 19:

Der Anmelder kann auf eigenen Wunsch die Ansprüche der internationalen Anmeldung ändern (siehe Regel 46):

Bis wann sind Änderungen einzureichen?

Die Frist zur Einreichung solcher Änderungen beträgt üblicherweise zwei Monate ab der Übermittlung des internationalen Recherchenberichts; weitere Einzelheiten sind den Anmerkungen auf dem Beiblatt zu entnehmen.

Wo sind Änderungen einzureichen?

Unmittelbar beim Internationalen Büro der WIPO, 34, CHEMIN des Colombettes, CH-1211 Genf 20,
Telefaxnr.: (41-22) 740.14.35

Nähere Hinweise sind den Anmerkungen auf dem Beiblatt zu entnehmen.

2. ☐ Dem Anmelder wird mitgeteilt, daß kein internationaler Recherchenbericht erstellt wird und daß ihm hiermit die Erklärung nach Artikel 17(2)a) übermittelt wird.
3. ☐ Hinsichtlich des Widerspruchs gegen die Entrichtung einer zusätzlichen Gebühr (zusätzlicher Gebühren) nach Regel 40.2 wird dem Anmelder mitgeteilt, daß
- ☐ der Widerspruch und die Entscheidung hierüber zusammen mit seinem Antrag auf Übermittlung des Wortlauts sowohl des Widerspruchs als auch der Entscheidung hierüber an die Bestimmungsämter dem Internationalen Büro übermittelt worden sind.
- ☐ noch keine Entscheidung über den Widerspruch vorliegt; der Anmelder wird benachrichtigt, sobald eine Entscheidung getroffen wurde.

4. **Weiteres Vorgehen:** Der Anmelder wird auf folgendes aufmerksam gemacht:

Kurz nach Ablauf von **18 Monaten** seit dem Prioritätsdatum wird die internationale Anmeldung vom Internationalen Büro veröffentlicht. Will der Anmelder die Veröffentlichung verhindern oder auf einen späteren Zeitpunkt verschieben, so muß gemäß Regel 90 bis bzw. 90^{bis} 3 vor Abschluß der technischen Vorbereitungen für die internationale Veröffentlichung eine Erklärung über die Zurücknahme der internationalen Anmeldung oder des Prioritätsanspruchs beim Internationalen Büro eingehen.

Innerhalb von **19 Monaten** seit dem Prioritätsdatum ist ein Antrag auf internationale vorläufige Prüfung einzureichen, wenn der Anmelder den Eintritt in die nationale Phase bis zu 30 Monaten seit dem Prioritätsdatum (in manchen Ämtern sogar noch länger) verschieben möchte.

Innerhalb von **20 Monaten** seit dem Prioritätsdatum muß der Anmelder die für den Eintritt in die nationale Phase vorgeschriebenen Handlungen vor allen Bestimmungsämtern vornehmen, die nicht innerhalb von 19 Monaten seit dem Prioritätsdatum in der Anmeldung oder einer nachträglichen Auswahlklärung ausgewählt wurden oder nicht ausgewählt werden konnten, da für sie Kapitel II des Vertrages nicht verbindlich ist.

Name und Postanschrift der Internationalen Recherchenbehörde



Europäisches Patentamt, P.B. 5818 Patentlaan 2
NL-2280 HV Rijswijk
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
Fax: (+31-70) 340-3016

Bevollmächtigter Bediensteter

Lucia Van Pinxteren

ANMERKUNGEN ZU FORMBLATT PCT/ISA/220

Diese Anmerkungen sollen grundlegende Hinweise zur Einreichung von Änderungen gemäß Artikel 19 geben. Diesen Anmerkungen liegen die Erfordernisse des Vertrags über die internationale Zusammenarbeit auf dem Gebiet des Patentwesens (PCT), der Ausführungsordnung und der Verwaltungsrichtlinien zu diesem Vertrag zugrunde. Bei Abweichungen zwischen diesen Anmerkungen und obengenannten Texten sind letztere maßgebend. Nähere Einzelheiten sind dem PCT-Leitfaden für Anmelder, einer Veröffentlichung der WIPO, zu entnehmen.

Die in diesen Anmerkungen verwendeten Begriffe "Artikel", "Regel" und "Abschnitt" beziehen sich jeweils auf die Bestimmungen des PCT-Vertrags, der PCT-Ausführungsordnung bzw. der PCT-Verwaltungsrichtlinien.

HINWEISE ZU ÄNDERUNGEN GEMÄSS ARTIKEL 19

Nach Erhalt des internationalen Recherchenberichts hat der Anmelder die Möglichkeit, einmal die Ansprüche der internationalen Anmeldung zu ändern. Es ist jedoch zu betonen, daß, da alle Teile der internationalen Anmeldung (Ansprüche, Beschreibung und Zeichnungen) während des internationalen vorläufigen Prüfungsverfahrens geändert werden können, normalerweise keine Notwendigkeit besteht, Änderungen der Ansprüche nach Artikel 19 einzureichen, außer wenn der Anmelder z.B. zum Zwecke eines vorläufigen Schutzes die Veröffentlichung dieser Ansprüche wünscht oder ein anderer Grund für eine Änderung der Ansprüche vor ihrer internationalen Veröffentlichung vorliegt. Weiterhin ist zu beachten, daß ein vorläufiger Schutz nur in einigen Staaten erhältlich ist.

Welche Teile der Internationalen Anmeldung können geändert werden?

Im Rahmen von Artikel 19 können nur die Ansprüche geändert werden.

In der internationalen Phase können die Ansprüche auch nach Artikel 34 vor der mit der internationalen vorläufigen Prüfung beauftragten Behörde geändert (oder nochmals geändert) werden. Die Beschreibung und die Zeichnungen können nur nach Artikel 34 vor der mit der internationalen vorläufigen Prüfung beauftragten Behörde geändert werden.

Beim Eintritt in die nationale Phase können alle Teile der internationalen Anmeldung nach Artikel 28 oder gegebenenfalls Artikel 41 geändert werden.

Bis wann sind Änderungen einzureichen?

Innerhalb von zwei Monaten ab der Übermittlung des internationalen Recherchenberichts oder innerhalb von sechzehn Monaten ab dem Prioritätsdatum, je nachdem, welche Frist später abläuft. Die Änderungen gelten jedoch als rechtzeitig eingereicht, wenn sie dem internationalen Büro nach Ablauf der maßgebenden Frist, aber noch vor Abschluß der technischen Vorbereitungen für die internationale Veröffentlichung (Regel 46.1) zugehen.

Wo sind die Änderungen nicht einzureichen?

Die Änderungen können nur beim Internationalen Büro, nicht aber beim Anmeldeamt oder der Internationalen Recherchenbehörde eingereicht werden (Regel 46.2).

Falls ein Antrag auf internationale vorläufige Prüfung eingereicht wurde/wird, siehe unten.

In welcher Form können Änderungen erfolgen?

Eine Änderung kann erfolgen durch Streichung eines oder mehrerer ganzer Ansprüche, durch Hinzufügung eines oder mehrerer neuer Ansprüche oder durch Änderung des Wortlauts eines oder mehrerer Ansprüche in der eingereichten Fassung.

Für jedes Anspruchsblatt, das sich aufgrund einer oder mehrerer Änderungen von dem ursprünglich eingereichten Blatt unterscheidet, ist ein Ersatzblatt einzureichen.

Alle Ansprüche, die auf einem Ersatzblatt erscheinen, sind mit arabischen Ziffern zu numerieren. Wird ein Anspruch gestrichen, so brauchen die anderen Ansprüche nicht neu numeriert zu werden. Im Fall einer Neunummerierung sind die Ansprüche fortlaufend zu numerieren (Verwaltungsrichtlinien, Abschnitt 205 b)).

Die Änderungen sind in der Sprache abzufassen, in der die internationale Anmeldung veröffentlicht wird.

Welche Unterlagen sind den Änderungen beizufügen?

Begleitschreiben (Abschnitt 205 b)):

Die Änderungen sind mit einem Begleitschreiben einzureichen.

Das Begleitschreiben wird nicht zusammen mit der internationalen Anmeldung und den geänderten Ansprüchen veröffentlicht. Es ist nicht zu verwechseln mit der "Erklärung nach Artikel 19(1)" (siehe unten, "Erklärung nach Artikel 19 (1)").

Das Begleitschreiben ist nach Wahl des Anmelders in englischer oder französischer Sprache abzufassen. Bei englischsprachigen internationalen Anmeldungen ist das Begleitschreiben aber ebenfalls in englischer, bei französischsprachigen internationalen Anmeldungen in französischer Sprache abzufassen.

ANMERKUNGEN ZU FORMBLATT PCT/ISA/220 (Fortsetzung)

Im Begleitschreiben sind die Unterschiede zwischen den Ansprüchen in der eingereichten Fassung und den geänderten Ansprüchen anzugeben. So ist insbesondere zu jedem Anspruch in der internationalen Anmeldung anzugeben (gleichlautende Angaben zu verschiedenen Ansprüchen können zusammengefaßt werden), ob

- i) der Anspruch unverändert ist;
- ii) der Anspruch gestrichen worden ist;
- iii) der Anspruch neu ist;
- iv) der Anspruch einen oder mehrere Ansprüche in der eingereichten Fassung ersetzt;
- v) der Anspruch auf die Teilung eines Anspruchs in der eingereichten Fassung zurückzuführen ist.

Im folgenden sind Beispiele angegeben, wie Änderungen im Begleitschreiben zu erläutern sind:

1. [Wenn anstelle von ursprünglich 48 Ansprüchen nach der Änderung einiger Ansprüche 51 Ansprüche existieren]:
"Die Ansprüche 1 bis 29, 31, 32, 34, 35, 37 bis 48 werden durch geänderte Ansprüche gleicher Numerierung ersetzt; Ansprüche 30, 33 und 36 unverändert; neue Ansprüche 49 bis 51 hinzugefügt."
2. [Wenn anstelle von ursprünglich 15 Ansprüchen nach der Änderung aller Ansprüche 11 Ansprüche existieren]:
"Geänderte Ansprüche 1 bis 11 treten an die Stelle der Ansprüche 1 bis 15."
3. [Wenn ursprünglich 14 Ansprüche existierten und die Änderungen darin bestehen, daß einige Ansprüche gestrichen werden und neue Ansprüche hinzugefügt werden]:
"Ansprüche 1 bis 6 und 14 unverändert; Ansprüche 7 bis 13 gestrichen; neue Ansprüche 15, 16 und 17 hinzugefügt. "Oder" Ansprüche 7 bis 13 gestrichen; neue Ansprüche 15, 16 und 17 hinzugefügt; alle übrigen Ansprüche unverändert."
4. [Wenn verschiedene Arten von Änderungen durchgeführt werden]:
"Ansprüche 1-10 unverändert; Ansprüche 11 bis 13, 18 und 19 gestrichen; Ansprüche 14, 15 und 16 durch geänderten Anspruch 14 ersetzt; Anspruch 17 in geänderte Ansprüche 15, 16 und 17 unterteilt; neue Ansprüche 20 und 21 hinzugefügt."

"Erklärung nach Artikel 19(1)" (Regel 46.4)

Den Änderungen kann eine Erklärung beigelegt werden, mit der die Änderungen erläutert und ihre Auswirkungen auf die Beschreibung und die Zeichnungen dargelegt werden (die nicht nach Artikel 19 (1) geändert werden können).

Die Erklärung wird zusammen mit der internationalen Anmeldung und den geänderten Ansprüchen veröffentlicht.

Sie ist in der Sprache abzufassen, in der die internationale Anmeldung veröffentlicht wird.

Sie muß kurz gehalten sein und darf, wenn in englischer Sprache abgefaßt oder ins Englische übersetzt, nicht mehr als 500 Wörter umfassen.

Die Erklärung ist nicht zu verwechseln mit dem Begleitschreiben, das auf die Unterschiede zwischen den Ansprüchen in der eingereichten Fassung und den geänderten Ansprüchen hinweist, und ersetzt letzteres nicht. Sie ist auf einem gesonderten Blatt einzureichen und in der Überschrift als solche zu kennzeichnen, vorzugsweise mit den Worten "Erklärung nach Artikel 19 (1)".

Die Erklärung darf keine herabsetzenden Äußerungen über den internationalen Recherchenbericht oder die Bedeutung von in dem Bericht angeführten Veröffentlichungen enthalten. Sie darf auf im internationalen Recherchenbericht angeführte Veröffentlichungen, die sich auf einen bestimmten Anspruch beziehen, nur im Zusammenhang mit einer Änderung dieses Anspruchs Bezug nehmen.

Auswirkungen eines bereits gestellten Antrags auf internationale vorläufige Prüfung

Ist zum Zeitpunkt der Einreichung von Änderungen nach Artikel 19 bereits ein Antrag auf internationale vorläufige Prüfung gestellt worden, so sollte der Anmelder in seinem Interesse gleichzeitig mit der Einreichung der Änderungen beim Internationalen Büro auch eine Kopie der Änderungen bei der mit der internationalen vorläufigen Prüfung beauftragten Behörde einreichen (siehe Regel 62.2 a), erster Satz).

Auswirkungen von Änderungen hinsichtlich der Übersetzung der internationalen Anmeldung beim Eintritt in die nationale Phase

Der Anmelder wird darauf hingewiesen, daß bei Eintritt in die nationale Phase möglicherweise anstatt oder zusätzlich zu der Übersetzung der Ansprüche in der eingereichten Fassung eine Übersetzung der nach Artikel 19 geänderten Ansprüche an die bestimmten/ausgewählten Ämter zu übermitteln ist.

Nähere Einzelheiten über die Erfordernisse jedes bestimmten/ausgewählten Amtes sind Band II des PCT-Leitfadens für Anmelder zu entnehmen.

**VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT
AUF DEM GEBIET DES PATENTWESENS**

PCT

INTERNATIONALER RECHERCHENBERICHT

(Artikel 18 sowie Regeln 43 und 44 PCT)

Aktenzeichen des Anmelders oder Anwalts GR 98P2167P	WEITERES VORGEHEN siehe Mitteilung über die Übermittlung des internationalen Recherchenberichts (Formblatt PCT/ISA/220) sowie, soweit zutreffend, nachstehender Punkt 5	
Internationales Aktenzeichen PCT/DE 99/ 01999	Internationales Anmeldedatum (Tag/Monat/Jahr) 01/07/1999	(Frühestes) Prioritätsdatum (Tag/Monat/Jahr) 27/07/1998
Anmelder SIEMENS AKTIENGESELLSCHAFT et al.		

Dieser internationale Recherchenbericht wurde von der Internationalen Recherchenbehörde erstellt und wird dem Anmelder gemäß Artikel 18 übermittelt. Eine Kopie wird dem Internationalen Büro übermittelt.

Dieser internationale Recherchenbericht umfaßt insgesamt 3 Blätter.



Darüber hinaus liegt ihm jeweils eine Kopie der in diesem Bericht genannten Unterlagen zum Stand der Technik bei.

1. Grundlage des Berichts

- a. Hinsichtlich der **Sprache** ist die internationale Recherche auf der Grundlage der internationalen Anmeldung in der Sprache durchgeführt worden, in der sie eingereicht wurde, sofern unter diesem Punkt nichts anderes angegeben ist.



Die internationale Recherche ist auf der Grundlage einer bei der Behörde eingereichten Übersetzung der internationalen Anmeldung (Regel 23.1 b)) durchgeführt worden.

- b. Hinsichtlich der in der internationalen Anmeldung offenbaren **Nucleotid- und/oder Aminosäuresequenz** ist die internationale Recherche auf der Grundlage des Sequenzprotokolls durchgeführt worden, das



in der internationalen Anmeldung in schriftlicher Form enthalten ist.



zusammen mit der internationalen Anmeldung in computerlesbarer Form eingereicht worden ist.



bei der Behörde nachträglich in schriftlicher Form eingereicht worden ist.



bei der Behörde nachträglich in computerlesbarer Form eingereicht worden ist.



Die Erklärung, daß das nachträglich eingereichte schriftliche Sequenzprotokoll nicht über den Offenbarungsgehalt der internationalen Anmeldung im Anmeldezeitpunkt hinausgeht, wurde vorgelegt.



Die Erklärung, daß die in computerlesbarer Form erfaßten Informationen dem schriftlichen Sequenzprotokoll entsprechen, wurde vorgelegt.

2. ☐ **Bestimmte Ansprüche haben sich als nicht recherchierbar erwiesen** (siehe Feld I).

3. ☐ **Mangelnde Einheitlichkeit der Erfindung** (siehe Feld II).

4. Hinsichtlich der Bezeichnung der Erfindung



wird der vom Anmelder eingereichte Wortlaut genehmigt.



wurde der Wortlaut von der Behörde wie folgt festgesetzt:

5. Hinsichtlich der Zusammenfassung



wird der vom Anmelder eingereichte Wortlaut genehmigt.



wurde der Wortlaut nach Regel 38.2b) in der in Feld III angegebenen Fassung von der Behörde festgesetzt. Der Anmelder kann der Behörde innerhalb eines Monats nach dem Datum der Absendung dieses internationalen Recherchenberichts eine Stellungnahme vorlegen.

6. Folgende Abbildung der **Zeichnungen** ist mit der Zusammenfassung zu veröffentlichen: Abb. Nr. —



wie vom Anmelder vorgeschlagen



weil der Anmelder selbst keine Abbildung vorgeschlagen hat.



weil diese Abbildung die Erfindung besser kennzeichnet.



keine der Abb.

A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES
 IPK 7 G06F17/30

Nach der Internationalen Patentklassifikation (IPK) oder nach der nationalen Klassifikation und der IPK

B. RECHERCHIERTE GEBIETE

Recherchierter Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)

IPK 7 G06F

Recherchierte aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
X	DATE C J: "DISTRIBUTED DATABASES", US, READING, ADDISON WESLEY, PAGE(S) 291-340-309 XP002016220 Seite 318, Zeile 7 -Seite 323, Zeile 15 ---	1-6, 12-17,23
X	ROTHNIE J B ET AL: "INTRODUCTION TO A SYSTEM FOR DISTRIBUTED DATABASES (SDD-1)" ACM TRANSACTIONS ON DATABASE SYSTEMS, US, ASSOCIATION FOR COMPUTING MACHINERY. NEW YORK, Bd. 5, Nr. 1, Seite 1-17 XP000747926 ISSN: 0362-5915 Seite 7, Zeile 33 -Seite 10, Zeile 15 ---	1-6, 12-17,23
A	EP 0 716 385 A (XEROX CORP) 12. Juni 1996 (1996-06-12) Spalte 5, Zeile 58 -Spalte 9, Zeile 11 --- -/--	1-8, 12-19,23

☒ Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen

☒ Siehe Anhang Patentfamilie

* Besondere Kategorien von angegebenen Veröffentlichungen :

"A" Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist

"E" älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist

"L" Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)

"O" Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht

"P" Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist

"T" Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist

"X" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfinderischer Tätigkeit beruhend betrachtet werden

"Y" Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann nahelegend ist

"&" Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche

11. Januar 2000

Absendedatum des internationalen Recherchenberichts

17/01/2000

 Name und Postanschrift der internationalen Recherchenbehörde
 Europäisches Patentamt, P.B. 5818 Patentlaan 2
 NL - 2280 HV Rijswijk
 Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,
 Fax: (+31-70) 340-3016

Bevollmächtigter Bediensteter

Abbing, R

C.(Fortsetzung) ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
A	<p>MUTH P ET AL: "SEMANTIC CONCURRENCY CONTROL IN OBJECT-ORIENTED DATABASE SYSTEMS" PROCEEDINGS OF THE INTERNATIONAL CONFERENCE ON DATA ENGINEERING,US,LOS ALAMITOS, IEEE COMP. SOC. PRESS, Bd. CONF. 9, Seite 233-242 XP000380184 ISBN: 0-8186-3570-3 das ganze Dokument</p> <p>-----</p>	1,9,12, 20

INTERNATIONALER RECHERCHENBERICHT

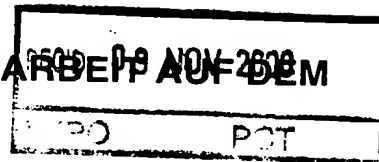
Angaben zu Veröffentlichungen, die zur selben Patentfamilie gehören

Internationales Aktenzeichen

PCT/DE 99/01999

Im Recherchenbericht angeführtes Patentdokument	Datum der Veröffentlichung	Mitglied(er) der Patentfamilie	Datum der Veröffentlichung
EP 0716385 A	12-06-1996	US 5671407 A	23-09-1997
		JP 2731373 B	25-03-1998
		JP 8241235 A	17-09-1996

VERTRAG ÜBER DIE INTERNATIONALE ZUSAMMENARBEIT AUF DEM GEBIET DES PATENTWESENS



PCT

INTERNATIONALER VORLÄUFIGER PRÜFUNGSBERICHT

(Artikel 36 und Regel 70 PCT)

Aktenzeichen des Anmelders oder Anwalts GR 98P2167P	WEITERES VORGEHEN siehe Mitteilung über die Übersendung des internationalen vorläufigen Prüfungsbericht (Formblatt PCT/IPEA/416)	
Internationales Aktenzeichen PCT/DE99/01999	Internationales Anmeldedatum (Tag/Monat/Jahr) 01/07/1999	Prioritätsdatum (Tag/Monat/Jahr) 27/07/1998
Internationale Patentklassifikation (IPK) oder nationale Klassifikation und IPK G06F17/30		
Anmelder SIEMENS AKTIENGESELLSCHAFT et al.		



- Dieser internationale vorläufige Prüfungsbericht wurde von der mit der internationale vorläufigen Prüfung beauftragte Behörde erstellt und wird dem Anmelder gemäß Artikel 36 übermittelt.
- Dieser BERICHT umfaßt insgesamt 4 Blätter einschließlich dieses Deckblatts.

☒ Außerdem liegen dem Bericht ANLAGEN bei; dabei handelt es sich um Blätter mit Beschreibungen, Ansprüchen und/oder Zeichnungen, die geändert wurden und diesem Bericht zugrunde liegen, und/oder Blätter mit vor dieser Behörde vorgenommenen Berichtigungen (siehe Regel 70.16 und Abschnitt 607 der Verwaltungsrichtlinien zum PCT).

 Diese Anlagen umfassen insgesamt 4 Blätter.

3. Dieser Bericht enthält Angaben zu folgenden Punkten:

- I ☒ Grundlage des Berichts
- II ☐ Priorität
- III ☐ Keine Erstellung eines Gutachtens über Neuheit, erfinderische Tätigkeit und gewerbliche Anwendbarkeit
- IV ☐ Mangelnde Einheitlichkeit der Erfindung
- V ☒ Begründete Feststellung nach Artikel 35(2) hinsichtlich der Neuheit, der erfinderische Tätigkeit und der gewerbliche Anwendbarkeit; Unterlagen und Erklärungen zur Stützung dieser Feststellung
- VI ☐ Bestimmte angeführte Unterlagen
- VII ☐ Bestimmte Mängel der internationalen Anmeldung
- VIII ☐ Bestimmte Bemerkungen zur internationalen Anmeldung

Datum der Einreichung des Antrags 18/02/2000	Datum der Fertigstellung dieses Berichts 07.11.2000
Name und Postanschrift der mit der internationalen vorläufigen Prüfung beauftragten Behörde:  Europäisches Patentamt D-80298 München Tel. +49 89 2399 - 0 Tx: 523656 epmu d Fax: +49 89 2399 - 4465	Bevollmächtigter Bediensteter Herry, T Tel. Nr. +49 89 2399 7406 

INTERNATIONALER VORLÄUFIGER PRÜFUNGSBERICHT

Internationales Aktenzeichen PCT/DE99/01999

I. Grundlage des Berichts

1. Dieser Bericht wurde erstellt auf der Grundlage (*Ersatzblätter, die dem Anmeldeamt auf eine Aufforderung nach Artikel 14 hin vorgelegt wurden, gelten im Rahmen dieses Berichts als "ursprünglich eingereicht" und sind ihm nicht beigelegt, weil sie keine Änderungen enthalten.*):

Beschreibung, Seiten:

1-52 ursprüngliche Fassung

Patentansprüche, Nr.:

1-23 mit Telefax vom 23/10/2000

Zeichnungen, Blätter:

1/4-4/4 ursprüngliche Fassung

2. Aufgrund der Änderungen sind folgende Unterlagen fortgefallen:

- ☐ Beschreibung, Seiten:
☐ Ansprüche, Nr.:
☐ Zeichnungen, Blatt:

3. ☐ Dieser Bericht ist ohne Berücksichtigung (von einigen) der Änderungen erstellt worden, da diese aus den angegebenen Gründen nach Auffassung der Behörde über den Offenbarungsgehalt in der ursprünglich eingereichten Fassung hinausgehen (Regel 70.2(c)):

4. Etwaige zusätzliche Bemerkungen:

V. Begründete Feststellung nach Artikel 35(2) hinsichtlich der Neuheit, der erfinderischen Tätigkeit und der gewerblichen Anwendbarkeit; Unterlagen und Erklärungen zur Stützung dieser Feststellung

1. Feststellung

Neuheit (N)	Ja: Ansprüche	1-23
	Nein: Ansprüche	
Erfinderische Tätigkeit (ET)	Ja: Ansprüche	1-23
	Nein: Ansprüche	
Gewerbliche Anwendbarkeit (GA)	Ja: Ansprüche	1-23
	Nein: Ansprüche	

**INTERNATIONALER VORLÄUFIGER
PRÜFUNGSBERICHT**

Internationales Aktenzeichen PCT/DE99/01999

2. Unterlagen und Erklärungen

siehe Beiblatt

Es wird auf Dokument D1: DE 196 07 132 A1 verwiesen, zitiert in der Anmeldung.

Der unabhängige Anspruch 1 genügt den Erfordernissen des Artikels 33(3) PCT.

Das Dokument D1 ist der nächstliegende Stand der Technik. Es beschreibt ein Verfahren zur Behebung von Inkonsistenzen in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, bei dem mindestens zwei Änderungsoperationen zu einer Folge von Operationen zusammengefaßt werden, die gemeinsam abgeglichen werden.

Das im Anspruch 1 definierte Verfahren unterscheidet sich vom in D1 beschriebenen Verfahren in folgenden Punkten: (a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen, ist definierten Konflikttypen zugeordnet; (b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann; (c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.

Technische Aufgabe: angeben eines weiteren Verfahrens zur Behebung von Inkonsistenzen in einer Datenbankmenge.

Erfinderische Tätigkeit: durch die Erfindung wird es möglich, eine Inkonsistenz in einer komplexen Datenbank generisch zu lösen. Durch das Zusammenfassen von möglichen Entscheidungen mit welchen ein spezifisches Problem (Inkonsistenz) behoben werden kann und durch das Zuordnen von Entscheidungssets zu vordefinierten Konflikttypen weist die Erfindung eine große Flexibilität auf (mehrere Aktionsarten möglich). In einer Ausgestaltung der Erfindung können diese Entscheidungssets abhängig von Abhängigkeiten der Inkonsistenzen verändert (eingeschränkt) werden, was weiterhin zu Rechenzeiteinsparung bei der Behebung von Inkonsistenzen beiträgt. Der Stand der Technik liefert keinen Hinweis auf Verfahren mit irgendeinem der Lösungsmerkmale (a), (b) oder (c).

Die unabhängigen Ansprüche 12 und 23 definieren die entsprechende Anordnung bzw. Satz mehrerer Anordnungen, und genügen den Erfordernissen des Art. 33(3) PCT auch.
Die abhängigen Ansprüche genügen den Erfordernissen des Art. 33(3) PCT auch.

Patentansprüche

1. Verfahren zur rechnergestützten Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht,
 - a) bei dem mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen, definierten Konflikttypen zugeordnet ist,
 - b) bei dem jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
 - c) bei dem die Inkonsistenz unter Verwendung des Entscheidungssets behoben wird.
2. Verfahren nach Anspruch 1,
3. Verfahren nach Anspruch 1 oder 2, bei dem jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mehrere Operationen des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
4. Verfahren nach einem der Ansprüche 1 bis 3, bei dem die Datenbankmenge mehrere Kopiedatenbanken der Datenbank aufweist.
5. Verfahren nach einem der Ansprüche 1 bis 4, bei dem vor der Behebung der Inkonsistenzen alle Inkonsistenzen und deren Abhängigkeiten voneinander ermittelt werden.
6. Verfahren nach einem der Ansprüche 1 bis 5,

bei dem bei der Ermittlung einer Inkonsistenz ein Konflikt, eine Anomalie oder eine Pseudo-Anomalie ermittelt wird.

7. Verfahren nach einem der Ansprüche 1 bis 6,
- 5 bei dem während der Behebung der Inkonsistenzen das Entscheidungsset mindestens eines Konflikttyps in Abhängigkeit von den Abhängigkeiten der Inkonsistenzen verändert wird.
8. Verfahren nach einem der Ansprüche 1 bis 7,
- 10 bei dem nach einer vorgebbaren Anzahl von behobenen Inkonsistenzen die Datenbankmenge auf weitere Inkonsistenzen und deren Abhängigkeiten, Anomalien und Pseudo-Anomalien hin untersucht wird.
- 15 9. Verfahren nach einem der Ansprüche 1 bis 8, bei dem die Datenbankmenge eine objektorientierte Datenbank enthält.
10. Verfahren nach einem der Ansprüche 1 bis 9, eingesetzt im Rahmen der objektorientierten Softwareentwicklung.
11. Verfahren nach einem der Ansprüche 1 bis 9, eingesetzt im Rahmen der Erstellung eines strukturierten
- 25 elektronischen Dokuments.
12. Anordnung zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht,
- 30 mit mindestens einem Prozessor, der derart eingerichtet ist, daß folgende Schritte durchführbar sind:
 - a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen, ist definierten Konflikttypen
 - 35 zugeordnet,

- b) jedem Konfliktyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konfliktyps erzeugte Inkonsistenz behoben werden kann,
- 5 c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.

13. Anordnung nach Anspruch 12,
bei der der Prozessor derart eingerichtet ist, daß mehrere
10 Inkonsistenzen behoben werden.

14. Anordnung nach Anspruch 12 oder 13,
bei der der Prozessor derart eingerichtet ist, daß jedem Kon-
fliktyp ein Entscheidungsset zugeordnet ist, mit dem mögli-
15 che Entscheidungen angegeben werden, mit denen eine durch
mehrere Operationen des jeweiligen Konfliktyps erzeugte In-
konsistenz behoben werden kann,

15. Anordnung nach einem der Ansprüche 12 bis 14,
20 bei der der Prozessor derart eingerichtet ist, daß die Daten-
bankmenge mehrere Kopiedatenbanken der Datenbank aufweist.

16. Anordnung nach einem der Ansprüche 12 oder 15,
bei der der Prozessor derart eingerichtet ist, daß vor der
25 Behebung der Inkonsistenzen alle Inkonsistenzen und deren Ab-
hängigkeiten voneinander ermittelbar sind.

17. Anordnung nach einem der Ansprüche 12 bis 16,
bei der der Prozessor derart eingerichtet ist, daß bei der
30 Ermittlung einer Inkonsistenz ein Konflikt, eine Anomalie
oder enie Pseudo-Anomalie ermittelbar ist.

18. Anordnung nach einem der Ansprüche 12 bis 17,
bei der der Prozessor derart eingerichtet ist, daß während
35 der Behebung der Inkonsistenzen das Entscheidungsset mindes-
tens eines Konfliktyps in Abhängigkeit von den Abhängigkei-
ten der Inkonsistenzen veränderbar ist.

19. Anordnung nach einem der Ansprüche 12 bis 18,
bei der der Prozessor derart eingerichtet ist, daß nach einer
vorgebbaren Anzahl von behobenen Inkonsistenzen die Daten-
bankmenge auf weitere Inkonsistenzen und deren Abhängigkei-
ten, Anomalien und Pseudo-Anomalien hin untersucht wird.
20. Anordnung nach einem der Ansprüche 12 bis 19,
bei der der Prozessor derart eingerichtet ist, daß die Daten-
bankmenge eine objektorientierte Datenbank enthält.
21. Anordnung nach einem der Ansprüche 12 bis 20,
eingesetzt im Rahmen der objektorientierten Softwareentwick-
lung.
22. Anordnung nach einem der Ansprüche 12 bis 21,
eingesetzt im Rahmen der Erstellung eines strukturierten
elektronischen Dokuments.
23. Satz mehrerer Anordnungen zur Behebung mindestens einer
Inkonsistenz in einer Datenbankmenge, die eine Datenbank so-
wie mindestens eine Kopiedatenbank der Datenbank aufweist,
welche Inkonsistenz durch Änderung der Datenbank und/oder der
Kopiedatenbank entsteht,
bei dem jede Anordnung mindestens einen Prozessor aufweist,
der derart eingerichtet ist, daß folgende Schritte durchführ-
bar sind:
- a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen, ist definierten Konflikttypen zugeordnet,
 - b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
 - c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben, und
- bei dem die Anordnungen miteinander koppelbar sind.

Patent claims

1. A method for the computer-aided elimination of at least one inconsistency in a database collection containing a database and at least one copy database of the database, which inconsistency arises on account of the database and/or the copy database being changed,
 - a) in which at least some of the operations which create an inconsistency are allocated to defined conflict types,
 - b) in which each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by at least one operation of the respective conflict type,
 - c) in which the inconsistency is eliminated using the decision set.
2. The method as claimed in claim 1, in which a plurality of inconsistencies are eliminated.
3. The method as claimed in claim 1 or 2, in which each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by a plurality of operations of the respective conflict type.
4. The method as claimed in one of claims 1 to 3, in which the database collection contains a plurality of copy databases of the database.
5. The method as claimed in one of claims 1 to 4, in which all the inconsistencies and their dependencies on one another are ascertained before the inconsistencies are eliminated.
6. The method as claimed in one of claims 1 to 5, in which a conflict, an anomaly or a pseudo-anomaly is ascertained when an inconsistency is ascertained.
7. The method as claimed in one of claims 1 to 6, in which, during elimination of the inconsistencies, the decision set for at least one conflict type is modified depending on the dependencies of the

- 51 -

inconsistencies.

8. The method as claimed in one of claims 1 to 7,
in which, after a prescribable number of eliminated
inconsistencies, the database collection is examined
5 for further inconsistencies and their dependencies,
anomalies and pseudo-anomalies.

9. The method as claimed in one of claims 1 to 8,
in which the database collection contains an object-
oriented database.

10 10. The method as claimed in one of claims 1 to 9,
used within the context of object-oriented software
development.

11. The method as claimed in one of claims 1 to 9,
used within the context of creating a structured
15 electronic document.

12. An arrangement for eliminating at least one
inconsistency in a database collection containing a
database and at least one copy database of the
database, which inconsistency arises on account of the
20 database and/or the copy database being changed,
having at least one processor which is set up such that
the following steps can be carried out:

- a) at least some of the operations which create an
inconsistency are allocated to defined conflict
25 types,
- b) each conflict type is allocated a decision set
which is used to indicate possible decisions which
can be used to eliminate an inconsistency created
by at least one operation of the respective
30 conflict type,
- c) the inconsistency is eliminated using the decision
set.

13. The arrangement as claimed in claim 12,
in which the processor is set up such that a plurality
35 of inconsistencies are eliminated.

14. The arrangement as claimed in claim 12 or 13,
in which the processor is set up such that each
conflict type is allocated a decision set which is used
to indicate possible decisions which can be used to

- 52 -

eliminate an inconsistency created by a plurality of operations of the respective conflict type.

15. The arrangement as claimed in one of claims 12 to 14,

5 in which the processor is set up such that the database collection contains a plurality of copy databases of the database.

16. The arrangement as claimed in one of claims 12 or 15,

10 in which the processor is set up such that all the inconsistencies and their dependencies on one another can be ascertained before the inconsistencies are eliminated.

15 17. The arrangement as claimed in one of claims 12 to 16,

in which the processor is set up such that a conflict, an anomaly or a pseudo-anomaly can be ascertained when an inconsistency is ascertained.

20 18. The arrangement as claimed in one of claims 12 to 17,

in which the processor is set up such that, during elimination of the inconsistencies, the decision set for at least one conflict type can be modified depending on the dependencies of the inconsistencies.

25 19. The arrangement as claimed in one of claims 12 to 18,

in which the processor is set up such that, after a prescribable number of eliminated inconsistencies, the database collection is examined for further
30 inconsistencies and their dependencies, anomalies and pseudo-anomalies.

20. The arrangement as claimed in one of claims 12 to 19,

35 in which the processor is set up such that the database collection contains an object-oriented database.

21. The arrangement as claimed in one of claims 12 to 20,

used within the context of object-oriented software development.

- 53 -

22. The arrangement as claimed in one of claims 12 to 21, used within the context of creating a structured electronic document.

- 5 23. A set of a plurality of arrangements for eliminating at least one inconsistency in a database collection containing a database and at least one copy database of the database, which inconsistency arises on account of the database and/or the copy database being
- 10 changed,
- in which each arrangement has at least one processor which is set up such that the following steps can be carried out:
- 15 a) at least some of the operations which create an inconsistency are allocated to defined conflict types,
- b) each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created
- 20 by at least one operation of the respective conflict type,
- c) the inconsistency is eliminated using the decision set, and
- in which the arrangements can be coupled to one
- 25 another.

REPLACED BY
ART 34 AMDT

Patent claims

1. A method for the computer-aided elimination of at least one inconsistency in a database collection
5 containing a database and at least one copy database of the database, which consistency arises on account of the database and/or the copy database being changed,
 - a) in which at least some of the operations which can create an inconsistency are allocated to defined
10 conflict types,
 - b) in which each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by at least one operation of the respective
15 conflict type,
 - c) in which the inconsistency is eliminated using the decision set.
2. The method as claimed in claim 1,
in which a plurality of inconsistencies are eliminated.
- 20 3. The method as claimed in claim 1 or 2,
in which each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by a plurality of operations of the respective conflict
25 type.
4. The method as claimed in one of claims 1 to 3,
in which the database collection contains a plurality of copy databases of the database.
5. The method as claimed in one of claims 1 to 4,
30 in which all the inconsistencies and their dependencies on one another, anomalies and pseudo-anomalies are ascertained before the inconsistencies are eliminated.
6. The method as claimed in one of claims 1 to 5,
in which the decision set for a conflict is modified
35 during the method.
7. The method as claimed in one of claims 1 to 6,
in which the respective decision set is changed on the basis of dependencies of inconsistencies.
8. The method as claimed in one of claims 1 to 7,

in which, after a prescribable number of eliminated inconsistencies, the database collection is examined for further inconsistencies and their dependencies, anomalies and pseudo-anomalies.

- 5 9. The method as claimed in one of claims 1 to 8, in which the database collection contains an object-oriented database.
- 10 10. The method as claimed in one of claims 1 to 9, used within the context of object-oriented software development.
11. The method as claimed in one of claims 1 to 9, used within the context of creating a structured electronic document.
- 15 12. An arrangement for eliminating at least one inconsistency in a database collection containing a database and at least one copy database of the database, which inconsistency arises on account of the database and/or the copy database being changed, having at least one processor which is set up such that
- 20 the following steps can be carried out:
- a) at least some of the operations which can create an inconsistency are allocated to defined conflict types,
 - b) each conflict type is allocated a decision set
 - 25 which is used to indicate possible decisions which can be used to eliminate an inconsistency created by at least one operation of the respective conflict type,
 - c) the inconsistency is eliminated using the decision
 - 30 set.
13. The arrangement as claimed in claim 12, in which the processor is set up such that a plurality of inconsistencies are eliminated.
14. The arrangement as claimed in claim 12 or 13,
- 35 in which the processor is set up such that each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created by a plurality of operations of the respective conflict type.

REPLACED BY
ART 34 AND 1

GR 98 P 2167

- 52 -

15. The arrangement as claimed in one of claims 12 to 14,
in which the processor is set up such that the database collection contains a plurality of copy databases of the database.
16. The arrangement as claimed in one of claims 12 and 15,
in which the processor is set up such that all the inconsistencies and their dependencies on one another, anomalies and pseudo-anomalies are ascertained before the inconsistencies are eliminated.
17. The arrangement as claimed in one of claims 12 to 16,
in which the processor is set up such that the decision set for a conflict is modified during the method.
18. The arrangement as claimed in one of claims 12 to 17,
in which the processor is set up such that the respective decision set is changed on the basis of dependencies of inconsistencies.
19. The arrangement as claimed in one of claims 12 to 18,
in which the processor is set up such that, after a prescribable number of eliminated inconsistencies, the database collection is examined for further inconsistencies and their dependencies, anomalies and pseudo-anomalies.
20. The arrangement as claimed in one of claims 12 to 19,
in which the processor is set up such that the database collection contains an object-oriented database.
21. The arrangement as claimed in one of claims 12 to 20,
used within the context of object-oriented software development.
22. The arrangement as claimed in one of claims 12 to 21,
used within the context of creating a structured electronic document.

REPLACED BY
ART 34 ANDT

GR 98 P 2167

- 53 -

23. A set of a plurality of arrangements for eliminating at least one inconsistency in a database collection containing a database and at least one copy database of the database, which inconsistency arises on account of the database and/or the copy database being changed,
- 5 in which each arrangement has at least one processor which is set up such that the following steps can be carried out:
- 10 a) at least some of the operations which can create an inconsistency are allocated to defined conflict types,
- b) each conflict type is allocated a decision set which is used to indicate possible decisions which can be used to eliminate an inconsistency created
- 15 by at least one operation of the respective conflict type,
- c) the inconsistency is eliminated using the decision set, and
- 20 in which the arrangements can be coupled to one another.

Beschreibung

5 **Verfahren, Anordnung und Satz mehrerer Anordnungen zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist**

10 Die Erfindung betrifft ein Verfahren, eine Anordnung sowie einen Satz mehrerer Anordnungen zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist.

15 Ein solches Verfahren ist aus [1] bekannt.

Bei dem aus [1] bekannten Verfahren kommunizieren Rechner über ein Kommunikationsnetz unter Verwendung eines Kommunikationsprotokolls miteinander.

20 Unter einem Kommunikationsnetz ist beispielsweise ein Daten-netz, ein Funknetz oder auch ein übliches Telefonnetz zu verstehen.

25 Unter einem Kommunikationsprotokoll ist ein Protokoll zur Festlegung des Datenformats zu verstehen, welches im Rahmen einer Kommunikation zwischen den Rechnern verwendet wird. Ein solches Kommunikationsprotokoll ist beispielsweise das Transport Control Protocol/Internet Protocol (TCP/IP).

30 Bei dem Verfahren aus [1] sind in einem ersten Rechner eine Datenbank und in jedem weiteren Rechner eine Kopie der Datenbank, im weiteren als Kopiedatenbank bezeichnet, gespeichert.

35 Die Datenbank bzw. die Kopiedatenbanken werden im Rahmen einer Sitzung von jeweils einem Rechner verändert, d.h. die in der Datenbank bzw. einer Kopiedatenbank enthaltenen Daten bzw. deren Struktur werden verändert.

Unter einer Datenbank ist in diesem Zusammenhang beispielsweise eine hierarchische oder auch eine objektorientierte Datenbank zu verstehen.

5

Eine Datenbank enthält Daten, die gemäß einer vorgegebenen Struktur gespeichert sind und miteinander in Zusammenhang stehen. Jedes Objekt, d.h. jeder Datensatz innerhalb der Datenbank ist üblicherweise über einen Identifikator

10 (Identifizier) eindeutig identifizierbar.

Es kommt vor, daß Änderungen an einer Kopiedatenbank vorgenommen werden, ohne daß dieselbe Änderung auch in der Datenbank selbst erfolgt oder auch umgekehrt.

15

Soll nun aus der jeweiligen Kopiedatenbank und der Datenbank eine konsistente Datenbank erstellt werden, so gilt es, eine durch Hinzufügen, Entfernen oder Ändern der Daten bzw. deren Struktur entstehende Inkonsistenz zu ermitteln und zu beheben.

20

Unter einer Inkonsistenz ist im weiteren jede syntaktische Differenz innerhalb einer Kopiedatenbank bzw. der Datenbank, d.h. alle in den Kopiedatenbanken bzw. der Datenbank auftretenden Abweichungen zwischen den in der Datenbank bzw. einer

25

Kopiedatenbank enthaltenen Datenelementen, ihren Eigenschaften sowie ihren Beziehungen zueinander zu verstehen.

In [1] sind verschiedene Möglichkeiten aufgezeigt, um eine solche Inkonsistenz zu beheben.

30

Der Erfindung liegt das Problem zugrunde, ein weiteres Verfahren bzw. eine weitere Vorrichtung zur Behebung von Inkonsistenzen in einer Datenbankmenge anzugeben, mit dem eine

35

möglichst Rechenzeit einsparende Behebung einer Inkonsistenz möglich wird.

Das Problem wird durch das Verfahren, durch die Anordnung sowie durch den Satz mehrerer Anordnungen mit den Merkmalen der unabhängigen Patentansprüche gelöst.

- 5 Das Verfahren zur rechnergestützten Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch eine Änderung in der Datenbank und/oder in der Kopiedatenbank entsteht, weist folgende
- 10 Schritte auf:
- a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
- 15 b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann, und
- 20 c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.

Die Anordnung zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Daten in der Datenbank oder in der Kopiedatenbank entsteht, weist mindestens einen Prozessor auf, der derart eingerichtet ist, daß folgende Schritte durchführbar sind:

- 30 a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
- b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine
- 35 durch eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,

c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.

- 5 Der Satz mehrerer Anordnungen zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht, weist mehrere Anordnungen auf, deren
- 10 jede mindestens einen Prozessor aufweist, der derart eingerichtet ist, daß folgende Schritte durchführbar sind:
- a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
- 15 b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
- c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.
- 20

Die Anordnungen sind miteinander koppelbar.

Durch die Erfindung wird es möglich, eine Inkonsistenz in einer komplexen Datenbank generisch zu lösen.

25

Bevorzugte Weiterbildungen der Erfindung ergeben sich aus den abhängigen Ansprüchen.

In einer bevorzugten Ausgestaltung werden mehrere Inkonsistenzen behoben.

30

Bevorzugt wird in einer weiteren Ausgestaltung jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch

35 mehrere Operationen des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann.

Ferner ist es in einer Weiterbildung vorgesehen, daß die Datenbankmenge mehrere Kopiedatenbanken der Datenbank aufweist.

5 Zur Vereinfachung und somit zur Rechenzeiteinsparung bei der Behebung einer Inkonsistenz ist es in einer Ausgestaltung der Erfindung vorgesehen, daß vor der Behebung jede Inkonsistenz und deren Abhängigkeiten voneinander ermittelt werden.

10 Eine weitere Einsparung an benötigter Rechenzeit zur Behebung mehrerer Inkonsistenzen wird in einer weiteren Ausgestaltung dadurch erreicht, daß das Entscheidungsset mindestens eines Konflikttyps während der Behebung der Inkonsistenzen verändert wird.

15 Dabei erfolgt die Änderung des jeweiligen Entscheidungssets bevorzugt abhängig von Abhängigkeiten der Inkonsistenzen.

20 In einer bevorzugten Ausgestaltung ist es vorgesehen, nach einer vorgebbaren Anzahl behobener Inkonsistenzen die Datenbankmenge auf weitere Inkonsistenzen und deren Abhängigkeit hin zu untersuchen.

25 Die Datenbankmenge enthält bevorzugt in einer Ausgestaltung eine objektorientierte Datenbank.

Das Verfahren kann im Rahmen der objektorientierten Softwareentwicklung oder auch im Rahmen der Erstellung eines strukturierten elektronischen Dokuments eingesetzt werden.

30 Ein Ausführungsbeispiel der Erfindung ist in den Figuren dargestellt und wird im weiteren näher erläutert.

Es zeigen

35 Figur 1 ein Ablaufdiagramm, in dem die Verfahrensschritte des Ausführungsbeispiels dargestellt sind;

Figur 2 eine Skizze, in der Rechner dargestellt sind, die über ein Kommunikationsnetz miteinander verbunden sind;

5 Figur 3 eine Skizze einer Datenbankstruktur;

Figur 4 eine Tabellenübersicht über mögliche Konflikte und Entscheidungssets mit Entscheidungsmöglichkeiten, um die jeweiligen Konflikte zu beheben.

10

Fig.2 zeigt einen ersten Rechner 200 mit einem Speicher 202 und einem Prozessor 203, die jeweils über einen Bus 204 miteinander und mit einer Eingangs-/Ausgangsschnittstelle 201 verbunden sind.

15

Über die Eingangs-/Ausgangsschnittstelle 201 ist der erste Rechner 200 mit einem Bildschirm 205, einer Tastatur 206 sowie einer Computermouse 207 verbunden.

20 Ferner ist der erste Rechner 200 über ein Kommunikationsnetz 260, in dem Beispiel ein ISDN-Netz (Integrated Services Digital Network) mit weiteren Rechnern 210, 220, 230, 240 und 250 verbunden.

25 In dem ersten Rechner 200 ist eine Datenbank 208 gespeichert.

Die weiteren Rechner 210, 220, 230, 240 und 250 weisen jeweils ebenfalls einen Prozessor 213, 223, 233, 243 und 253 sowie jeweils einen Speicher 212, 222, 232, 242 und 252 auf.

30 Jeweils der Prozessor 213, 223, 233, 243 und 253 und der Speicher 212, 222, 232, 242 und 252 sind über jeweils einen Bus 214, 224, 234, 244 und 254 über eine Eingangs-

35 /Ausgangsschnittstelle 211, 221, 231, 241 und 251 mit dem Kommunikationsnetz 260 verbunden. Ferner sind die weiteren Rechner 210, 220, 230, 240 und 250 jeweils mit einem Bildschirm 215, 225, 235, 245 und 255 sowie einer Tastatur 216,

226, 236, 246 und 256 sowie einer Computermouse 217, 227, 237, 247 und 257 verbunden.

5 Jeweils eine Kopie der Datenbank 208, im weiteren als Kopie-
datenbank 218, 228, 238, 248 und 258 bezeichnet, wird von dem
ersten Rechner 200 an jeweils einen weiteren Rechner 210,
220, 230, 240 und 250 übermittelt und dort in dessen Speicher
212, 222, 232, 242 und 252 gespeichert.

10 Nach Übermittlung der Kopiedatenbanken 218, 228, 238, 248 und
258 unterbrechen die Rechner 200, 210, 220, 230, 240 und 250
die Kommunikation und es erfolgt jeweils unter den Rechnern
Rechner 200, 210, 220, 230, 240 und 250 autark eine Änderung,
d.h. Entfernung oder Hinzufügung von Daten oder Entfernung
15 oder Hinzufügung von Abhängigkeiten der Daten in einer Kopie-
datenbank 218, 228, 238, 248 und 258 bzw. der Datenbank 208.

Nach Wiederaufnahme der Kommunikation zwischen dem ersten
Rechner 200 und den weiteren Rechnern 210, 220, 230, 240 und
20 250 soll eine konsistente Datenbank aus der Datenbank 208 und
den Kopiedatenbanken 218, 228, 238, 248 und 258 gebildet wer-
den.

25 Zu diesem Zweck ist es erforderlich, jeweils vorgenommene An-
derungen in der Datenbank 208 oder den Kopiedatenbanken fest-
zustellen, um somit Inkonsistenzen zwischen den Kopiedaten-
banken sowie der Datenbank 208 zu ermitteln, damit die Inkon-
sistenzen behoben werden können.

30 Unabhängig von der syntaktischen Struktur und der Abhängig-
keiten der Datenelemente untereinander kann jedes Datenele-
ment beliebig viele Eigenschaften besitzen. Jede Eigenschaft
ist dabei von einem bestimmten Eigenschaftstyp und wird durch
einen aktuellen Wert repräsentiert. Für alle Eigenschaften
35 wird bezüglich der Wertebereiche die Annahme getroffen, daß
alle Werte nur aus Symbolen oder zusammengesetzten Symbolen
einer ASCII-Tabelle bestehen dürfen (Ziffern, Zahlen, Buch-

staben, Sonderzeichen, Zeichenketten). Eine Folge solcher Zeichen und Symbole wird nachfolgend als Eintrag bezeichnet. Komplexere Eigenschaften werden bei der Anwendungsmodellierung durch Datenelemente und Beziehungen repräsentiert.

5

Im weiteren werden drei Typen von Eigenschaften bei der syntaktischen Analyse in Abhängigkeit der auf der Eigenschaft ausführbaren Operationen unterschieden:

10 • **Einzelner Wert:**

Ein „Wert“ als Eigenschaftstyp beschreibt einen einzelnen Eintrag, wobei der Eintrag immer in seiner Gesamtheit gesehen und auch so verändert wird. Eine Veränderung der Eigenschaft vom Typ „Wert“ erfolgt dabei immer durch eine vollkommene Ersetzung des Eintrages der Eigenschaft durch einen neuen Eintrag.

15

• **Aufzählung:**

Eine „Aufzählung“ als Eigenschaftstyp beschreibt eine Menge beliebiger Einträge, wobei die Einträge in keiner Relation zueinander stehen und ihrerseits einen einzelnen Wert, eine Aufzählung oder eine geordnete Aufzählung darstellen können. Die einzelnen Einträge können dabei nur einzeln hinzugefügt oder gelöscht werden. Die Eindeutigkeit der Einträge muß bei eventueller Anforderung durch die Anwendung gewährleistet werden. Ein Beispiel für eine Datenstruktur, die diesen Eigenschaftstyp repräsentiert ist eine Hash-Tabelle oder ein Array.

20

25

• **Geordnete Aufzählung:**

Eigenschaften des Typs „geordnete Aufzählung“ beschreiben wie Eigenschaften des einfachen Aufzählungstyps eine Menge beliebiger Einträge. Die Einträge stehen jedoch hier in einer definierten Reihenfolge zueinander, die über einen Index für jeden Eintrag festgelegt ist. Die Festlegung der Indices erfolgt relativ zum Beginn der Aufzählung. Eine Einfüge-Operation mit einzelnen oder mehreren Einträgen bezieht sich deshalb immer auf einen Index. Eine Löschoption kann sich auf einen einzelnen Eintrag mit nur ei-

30

35

nem Index oder auf eine Reihe aufeinanderfolgender Einträge und somit einem Anfangs- und einem Endindex beziehen. Das Kriterium der Reihenfolge wird von einem Anwendungsprogramm definiert und auch die Einhaltung der Reihenfolgekriterien wird von ihm überwacht. Ein Beispiel für diesen Eigenschaftstyp ist eine indizierte Liste mit beliebigen Einträgen (z.B. Textdokument), bei der jede Zeile oder jedes Zeichen einem Eintrag entspricht.

10 Unter einem Datenelement DE ist ein 4-Tupel zu verstehen, welches folgendermaßen definiert ist:

Datenelement

Ein Datenelement DE ist ein 4-Tupel

15

DE def (ID, inforaum, elementtyp, eigenschaften);

- ID ist ein systemweit eineindeutiger Identifier
- inforaum \in MIR; wobei MIR eine Menge aller Informationsräume ist
- elementtyp \in ET; wobei ET eine Menge aller Datenelementtypen ist
- eigenschaften $\subseteq \{(name, eigenschaftstyp, wert):$
- name \in MEN, wert \in MEW, eigenschaftstyp \in MET},
- 25 wobei:
- MEN eine Menge aller Eigenschaftsnamen ist und gilt:

$\forall i \in \{1, \dots, n\}; \forall k \in \{1, \dots, m\}: name_i \neq name_k,$

- 30 MEW eine Menge aller Eigenschaftswerte ist sowie
- MET eine Menge aller Eigenschaftstypen
- {„wert“, „aufzählung“, „geordnete aufzählung“} ist.

Ein Informationsraum wird im weiteren folgendermaßen definiert:

35

Informationsraum

Ein Informationsraum IR ist ein 3-Tupel
 IR def (ID, irname, eigentümer, daten)

- ID ist ein systemweit eineindeutiger Identifier,
 5 wobei gilt:
 $\forall i \in (1, \dots, n); \forall k \in (1, \dots, n); i \neq k:$
 $IR_i.ID \neq IR_k.ID;$
 wobei MIR eine Menge aller im System vorhandenen Irs ist
 und n deren Anzahl;
- 10 • irname \in MIRN
 wobei gilt:
 $\forall i \in (1, \dots, m); \forall k \in (1, \dots, m); i \neq k:$
 $Iri.irname \neq Irk.irname;$
 wobei MIR die Menge aller im System vorhandenen Irs ist und
 15 m deren Anzahl;
 MIRN eine Menge aller möglichen Informationsraumnamen dar-
 stellt;
- eigentümer = N_i mit $N_i \in MN$ oder N_{gi} mit $N_{gi} \in MNG;$
- daten sind die durch eine Nutzergruppe zugreifbaren und dem
 20 IR zugeordneten Daten.

Eine Beziehung zwischen den Datenelementen wird im weiteren
 folgendermaßen definiert:

25 **Beziehung zwischen Datenelementen**

Eine Beziehung BZ zwischen Datenelementen ist ein 3-Tupel

BZ def (beziehungstyp, name, datenelement1, datenelement2)

- 30 • name \in MBN; wobei MBN eine Menge aller Beziehungsnamen ist
- beziehungstyp \in MBT; wobei MBT eine Menge aller Beziehung-
 stypen
 $\{„ungerichtet“, „logisch“, „nachfolger“, „subsup“\}$ ist
- datenelement1, 2 \in MDE; wobei MDE eine Menge aller Da-
 35 tenelemente ist.

Um Inkonsistenzen zu ermitteln, wird in jedem Rechner 200, 210, 220, 230, 240 und 250 jeweils ein Protokoll über alle an der Datenbank bzw. der jeweiligen Kopiedatenbank vorgenommenen Operation mitgeführt und in Form einer Liste gespeichert.

5

Die gespeicherte Liste wird im weiteren als Historie bezeichnet.

10

Somit ist der Datenbank 208 sowie jeder Kopiedatenbank 218, 298, 238, 248 und 258 jeweils eine Historie zugeordnet.

15

Diese Situation ist in Fig.3 dargestellt. Fig.3 zeigt die Datenbank 301 mit Objekten 302, 303, 304 und 305 sowie einer Historie 306, die als Einträge 307, 308, 309 Änderungsoperationen gespeichert hat, die seit Unterbrechung der Kommunikation mit den weiteren Rechnern 210, 220, 230, 240 und 250 von dem ersten Rechner 200 an der Datenbank 301 durchgeführt wurden. Die Einträge 307, 308, 309, werden ebenfalls in dem Speicher 202 des ersten Rechners 200 gespeichert.

20

Einer ersten Kopiedatenbank 310 mit Objekten 311, 312, 313 und 314 ist ebenfalls eine Historie 315 mit entsprechenden Änderungsoperationen 316, 317, 318 zugeordnet. Die Kopiedatenbank 310 ist in dem weiteren Rechner 210 gespeichert.

25

Eine zweite Kopiedatenbank 320 mit Objekten 321, 322, 323 und der ihr zugeordneten Historie 325 mit Änderungsoperationen 326, 327, 328 ist in einem weiteren Rechner 220 gespeichert.

30

Zur Bildung der konsistenten Datenbank, d.h. zur Reintegration aller Kopiedatenbanken 218, 228, 238, 248 und 258 mit der Datenbank 208 werden die Historien 315, 325, ... zu dem ersten Rechner 200 über das Kommunikationsnetz 260 übertragen und in dem Speicher 202 des ersten Rechners 200 gespeichert.

35

Zu Beginn der Reintegration, der in Fig.1 durch Schritt 101 beschrieben ist, werden alle Historien der Kopiedatenbanken

zu dem ersten Rechner übertragen und dort gespeichert
(Schritt 102).

In einem dritten Schritt (Schritt 103) werden alle im Rahmen
5 der Reintegration zu berücksichtigende Historien 315, 325,
... bestimmt.

Im weiteren werden folgende Änderungsoperationen berücksich-
tigt, mit denen eindeutig die aufgetretenen Inkonsistenzen
10 beschrieben werden.

Im Rahmen dieses Ausführungsbeispiels werden folgende neun
Operationen als Änderungsoperationen berücksichtigt, die im
weiteren in Form eines Pseudo-Programmcodes beschrieben wer-
15 den:

1. Create Element:

Create Element (R(IR), ID, Elementtyp) → R(IR)

```
20 createElement(ir, id, elementtyp) RETURN R(IR)
    BEGIN element := instantiate(elementtyp)
        element.elementname:= id
        R(ir):= insert(R(ir).daten,element)
        R(ir):= add(R(ir).historie,
25      „createElement(id,elementtyp)“)
        return R(ir)
    END
```

Diese Operation erzeugt ein Datenelement vom Datentyp **ele-**
30 **menttyp** mit dem Identifikator **id** in einer Kopiedatenbank
R(ir) innerhalb eines Informationsraums **ir**, auf den diese
Operation angewendet wird. Dabei erhalten alle Eigenschaften
des neu erzeugten Datenelementes einen vorgesetzten Initiali-
sierungswert. Das neue Element wird nach dessen Initialisie-
35 rung unter dem angegebenen Namen zu den Daten der Kopiedaten-
bank **R(ir)** und die ausgeführte Operation ohne den Informati-
onsraum als Parameter in die der Kopiedatenbank zugeordneten

Historie **R(ir).historie** hinzugefügt. Die Erzeugung des eindeutigen Identifiers **id** ist dabei von der die Operation versendenden Anwendung vorzunehmen.

5 2. DeleteElement:

DeleteElement(**R(IR)**, **ID**) → **R(IR)**

deleteElement(**ir**, **id**) RETURN **R(IR)**

BEGIN **element** := select (**R(ir).daten**, **id**)

10 **R(ir)** := remove(**R(ir).daten**, **element**)

R(ir) := add(**R(ir).historie**, "deleteElement(**id**)")

return **R(ir)**

END

15 Diese Operation löscht ein Datenelement mit dem Namen **id** aus der Kopiedatenbank **R(ir)** des Informationsraums **ir** und schreibt die ausgeführte Operation in die der Kopiedatenbank zugeordneten Historie **R(ir).historie**. Alle seit der Instantiierung des Datenelements veränderten Eigenschaften des Datenelementes gehen dabei mit verloren. Die das Element betreffenden Beziehungen bleiben jedoch bestehen. Sollen diese ebenfalls gelöscht werden, so ist das Anwendungsprogramm dafür verantwortlich.

25 3. ChangeEigenschaft:

ChangeEigenschaft(**R(IR)**, **ID**, **Eigenschaftstyp**, **Wert**) → **R(IR)**

changeEigenschaft(**ir**, **id**, **eigname**, **neuerWert**) RETURN **R(IR)**

BEGIN **element** := select (**R(ir).daten**, **id**)

30 **eigenschaft** := select(**element.eigenschaften**,
eigname)

eigenschaft.wert := **neuerWert**

R(ir) := add (**R(ir).historie**, „changeEigenschaft
(**id**, **eigname**, **neuerWert**)")

35 return **R(ir)**

END

Diese Operation setzt in der Kopiedatenbank $R(ir)$ des Informationsraums ir den Wert der Eigenschaft **eigname** des Datenelementes mit dem Identifier id auf den Wert **neuerWert** und schreibt die ausgeführte Operation in die der Kopiedatenbank zugeordneten Historie $R(ir).historie$.

4. ChangeEigenschaftAdd:

$ChangeEigenschaftAdd(R(IR), ID, Eigenschaftstyp, Eintrag) \rightarrow R(IR)$

```

10  changeEigenschaftAdd(ir,id,eigname,neuerEintrag) RETURN R(IR)
      BEGIN element := select (R(ir).daten, id)
          eigenschaft := select (element.eigenschaften,
          eigname)
          add (eigenschaft.wert, neuerEintrag)
15  R(ir) := add(R(ir).historie,
          „changeEigenschaftAdd (id, eigname, neuerEin-
          trag)“)
          return R(ir)
      END

```

20

Diese Operation ist für Eigenschaften des Typs „aufzählung“. Die Operation fügt in der Kopiedatenbank $R(ir)$ des Informationsraums ir im Wert der Eigenschaft **eigname** des Datenelementes mit dem Identifier id am Ende der Aufzählung einen neuen Eintrag **neuerEintrag** hinzu. Anschließend wird die ausgeführte Operation in die der Kopiedatenbank zugeordneten Historie $R(ir).historie$ gespeichert.

25

5. ChangeEigenschaftDel:

ChangeEigenschaftDel(**R**(**IR**), **ID**, **Eigenschaftstyp**, **Index**,
Eintrag) → **R**(**IR**)

```
5  changeEigenschaftDel(ir, id, eigname, index, alterEintrag)  
    RETURN R(IR)  
    BEGIN element := select (R(ir).daten, id)  
        eigenschaft := select (element.eigenschaften,  
10      eigname)  
        del(eigenschaft.wert, alterEintrag)  
        R(ir) := add(R(ir).historie,  
        „changeEigenschaftDel(id, eigname, index, al-  
        terEintrag)“)  
        return R(ir)  
15      END
```

Diese Operation ist für Eigenschaften des Typs „aufzählung“. Die Operation löscht in der Kopiedatenbank **R(ir)** des Informationsraums **ir** im Wert der Eigenschaft **eigname** des Datenelementes mit dem Identifier **id** den ersten in der Aufzählung auftretenden Eintrag **alterEintrag**. Danach wird die ausgeführte Operation in der der Kopiedatenbank zugeordneten Historie **R(ir).historie** gespeichert.

6. ChangeEigenschaftInsert:

ChangeEigenschaftInsert(R(IR), ID, Eigenschaftstyp, Index,
Anzahl, Einträge) → R(IR)

```

5  changeEigenschaftInsert(ir, id, eigname, index, anzahl,
                             neueEinträge) RETURN R(IR)
      BEGIN element := select (R(ir).daten, id)
            eigenschaft := select (element.eigenschaften,
            eigname)
10      for (i=0, i < anzahl, i++){
            incrIndex (eigenschaft.wert.einträge, ind >= index)
            insert (eigenschaft.wert.index, neueEinträge.(anzahl - i))
15      R(ir) := add (R(ir).historie,
            „changeEigenschaftInsert (id, eigname, index, anzahl,
            neueEinträge)“)
            return R(ir)
      END

```

20

Eine Eigenschaft, auf die diese Operation angewendet werden kann, ist vom Typ „geordnete aufzählung“. Diese Operation fügt in der Kopiedatenbank **R(ir)** des Informationsraumes **ir** im Wert der Eigenschaft **eigname** des Datenelementes mit dem Identifier **id** ab der Position **Index** in der geordneten Aufzählung **Eigenschaft.Wert** eine Anzahl **anzahl** neuer Einträge **neueEinträge.i** ein. Alle Einträge der geordneten Aufzählung **eigenschaft.Wert** mit einem gleichen oder größeren Index als **index** wird **index** um den Wert **anzahl** erhöht. Danach wird die ausgeführte Operation in der der jeweilige Kopiedatenbank zugeordneten Historie **R(ir).historie**, gespeichert.

30

7. ChangeEigenschaftRemove:

ChangeEigenschaftRemove(R(IR), ID, Eigenschaftstyp, Index,
Anzahl, Einträge) → R(IR)

```

5  changeEigenschaftRemove(ir, id, eigname, index, anzahl,
    alteEinträge) RETURN R(IR)
    BEGIN element := select (R(ir).daten,id)
      eigenschaft := select (element.eigenschaften,
10      eigname)
      for (i = 0, i < anzahl, i++){
        remove (eigenschaft.wert.index, alteEinträge.
          (i+1))
        decrIndex (eigenschaft.wert.einträge, ind > in-
15      dex)}
      R(ir) := add(R(ir).historie,
        „changeEigenschaftRemove(id, eigname, index, an-
          zahl, alteEinträge)“)
      return R(ir)
    END

```

20

Eine Eigenschaft, auf die diese Operation angewendet werden kann, ist vom Typ „geordnete aufzählung“. Diese Operation löscht in der Kopiedatenbank R(ir) des Informationsraumes ir im Wert der Eigenschaft eigname des Datenelementes mit dem

25 Identifizier id ab der Position Index in der geordneten Aufzählung Eigenschaft.Wert die Einträge alteEinträge. Alle Einträge mit größerem Index als (index + anzahl) werden um den Wert anzahl in ihrem Index verringert. Danach wird die ausgeführte Operation in der der jeweiligen Kopiedatenbank zugeordneten

30 Historie R(ir).historie gespeichert.

8. CreateBeziehung:

CreateBeziehung(R(IR), Name, Beziehungstyp, ID1, ID2) → R(IR)

createBeziehung(ir, name, beztyp, fromid, toid, toidir)

```

5          RETURN R(IR)
      BEGIN beziehung := instantiate (beztyp)
            beziehung.name := name
            beziehung.datenelement1 := fromid
            beziehung.datenelement2 := toid
10         beziehung.datenelement2.ir := toidir
            R(ir) := insert (R(ir).daten, beziehung)
            R(ir) := add (R(ir).historie,
            „createBeziehung (name, beztyp, fromid, toid,
            toidir)“)
15         return R(ir)
      END

```

Diese Operation erzeugt eine Beziehung des Typs **beztyp** zwischen den Datenelementen mit den Identifiern **fromid** and **toid** unter dem Namen **name** und fügt die neue Beziehung zu den Daten der Kopiedatenbank **R(ir)** des Informationsraums **ir** hinzu. Danach wird die ausgeführte Operation in der der Kopiedatenbank zugeordneten Historie **R(ir).historie** gespeichert. Es wird für alle Beziehungen angenommen, daß es pro vergebenen Beziehungs-
 25 hungsnamen nur eine einzige Beziehung des gleichen Typs zwischen zwei Datenelementen gibt. Sind mehrere Beziehungen des gleichen Typs unter dem gleichen Namen zwischen zwei Datenelementen notwendig, so sind auch für Beziehungen Identifier einzuführen. Für die Mehrzahl der Anwendungen reicht die
 30 getroffene Annahme jedoch aus. Die Angabe des Informationsraumes des Zieldatenelementes ist nur im Falle einer logisch extern gerichteten Beziehung notwendig.

9. DeleteBeziehung:

DeleteBeziehung(R(IR), Name, Beziehungstyp, ID1, ID2)→R(IR)

deleteBeziehung(ir, name, beztyp, fromid, toid.toidir)

5

RETURN R(IR)

BEGIN beziehung := select (R(ir).daten, name, beztyp,
fromid, toid)

R(ir) := remove (R(ir).daten, beziehung)

R(ir) := add (R(ir).historie,

10

„deleteBeziehung (name, beztyp, fromid,
toid.toidir)“)

return R(ir)

END

- 15 Diese Operation löscht eine Beziehung des Typs **beztyp** zwischen den Datenelementen mit den Identifiern **fromid** and **toid** unter dem Name **name** aus der Kopiedatenbank **R(ir)** des Informationsraumes **ir**. Danach wird die ausgeführte Operation in der der Kopiedatenbank zugeordneten Historie **R(ir).historie** gespeichert. Die Angabe des Informationsraums des Zieldatenelements ist nur für die Beziehungen vom Typ logisch extern gerichtet notwendig.
- 20

- In einem weiteren Schritt werden alle Konflikte, Abhängigkeiten, Anomalien, Pseudo-Anomalien, sowie Einschränkungen durch Abhängigkeiten erkannt (Schritt 103).
- 25

- Unter einem Konflikt ist die kleinste entscheidbare Menge von syntaktisch nur einseitig auftretenden Operationen, die eine Inkonsistenz eindeutig beschreiben und einem Nutzer oder dem System sinnvoll präsentiert und von ihm behoben (entschieden) werden können, zu verstehen.
- 30

- Jeder Konflikt ist als Ganzes zu erkennen und durch eine einzige Entscheidung während der Reintegration lösbar.
- 35

Mögliche Konflikte werden anschließend abhängig von der Datenstruktur und den in der Historie vorkommenden Operationen definiert.

- 5 Es werden harmlose Konflikte und kritische Konflikte unterschieden.

Harmlose Konflikte (HK) beinhalten nur Operationen, die Veränderungen auf einer Kopiedatenbank beschreiben. Es gibt in diesem Fall somit nur einen Nutzer, der eine Änderung an dem Teil der Datenstruktur bzw. der Kopiedatenbank oder auch der Datenbank selbst wünscht und durchführt. Die vorgenommenen Operationen ergänzen sich somit. Abhängig davon, welcher Kopiedatenbank die Operation bzw. Operationen zuzuordnen sind, können allgemein bei Vorhandensein der Kopiedatenbanken von Benutzern A und B harmlose Konflikte mit Operationen auf der Kopiedatenbank eines ersten Benutzers A als HKA und harmlose Konflikte mit Operationen auf der Kopiedatenbank eines zweiten Benutzers B als HKB bezeichnet werden.

20

Kritische Konflikte (KK) dagegen enthalten beidseitige Änderungen zum gleichen Teil der Datenstruktur und stellen konträre Ansichten der Benutzer über den letztendlichen Zustand bestimmter Daten innerhalb der Datenbank bzw. Kopiedatenbanken dar. Dabei kann ein kritischer Konflikt auch durch unterschiedliche Operationen auf zwei Kopiedatenbanken definiert werden. In diesen Fällen wird zwischen einem kritischen Konflikt KKA und einem kritischen Konflikt KKB unterschieden.

25
30 Formal wird ein Konflikt wie folgt definiert:

Konflikt:

Ein Konflikt K zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 6-Tupel

5 K(EHA, EHB, GH) def

(id, ktyp, operationenEHA, operationenEHB,
operationenGH, entscheidungseinschr);

- id ist ein systemweit eindeutiger Identifier (siehe auch
10 die Definition eines Datenelements)
- ktyp \in {HK1A, ..., HK11A, HK1B, ..., HK11B, KK1, KK2,
KK3A, KK3B, KK4, KK5A, KK5B, KK6, KK7, KK8A, KK8B}
- operationenEHA \in EHA.operationen;
- operationenEHB \in EHB.operationen;
- 15 • operationenGH \in GH.operationen;
- entscheidungseinschr \subseteq MENTktyp; wobei MENTktyp eine Menge
aller möglichen Entscheidungen für einen Konflikt vom Typ
ktyp ist.

20 Die Konflikte sind in Fig.4 dargestellt.

1. Erster Harmloser Konflikt HK1:

HK1 = (createElement/ --)

Es liegt eine Erzeugungsoption eines Datenelements

25 createElement(id, elementtyp) in nur einer Historie vor.

HK1A = createElement(id, elementtyp) \in EHA \vee

HK1B = createElement(id, elementtyp) \in EHB.

2. Zweiter Harmloser Konflikt HK2:

30 HK2 = (deleteElement/ --)

Es liegt eine Löschoperation zu einem Datenelement

deleteelement(id, elementtyp) in nur einer Historie vor.

HK2A = deleteElement(id, elementtyp) \in EHA \vee

HK2B = deleteElement(id, elementtyp) \in EHB.

35

3. Dritter Harmloser Konflikt HK3:

HK3 = (createBeziehung/ --)

Es liegt eine Erzeugungsoperation einer Beziehung

createBeziehung (beztyp, bname, id1, id2) in nur einer Historie vor.

HK3A = createBeziehung(beztyp, bname, id1, id2) \in EHA \vee

HK3B = createBeziehung(beztyp, bname, id1, id2) \in EHB.

4. Vierter Harmloser Konflikt HK4:

HK4 = (deleteBeziehung/ --)

Es liegt eine Löschoption einer Beziehung

deleteBeziehung(beztyp, bname, id1, id2) in nur einer Historie vor.

HK4A = deleteBeziehung(beztyp, bname, id1, id2) \in EHA \vee

HK4B = deleteBeziehung(beztyp, bname, id1, id2) \in EHB.

5. Fünfter Harmloser Konflikt HK5:

HK5 = (deleteBeziehung12, createBeziehung 13/ --)

Es liegt eine Löschoption einer Beziehung

deleteBeziehung(beztyp, bname id1, id2) und eine nachfolgende Erzeugungsoperation der Beziehung

createBeziehung(beztyp, bname, id1, id3) vom gleichen Quell-datenelement zu einem anderen Zieldatenelement nur einer Historie vor.

HK5A = [deleteBeziehung(beztyp, bname, id1, id2),
createBeziehung(beztyp, bname, id1, id3)] \in EHA \vee

HK5B = [deleteBeziehung(beztyp, bname, id1, id2),
createBeziehung(beztyp, bname, id1, id3)] \in EHB.

6. Sechster Harmloser Konflikt HK6:

HK6 = (changeEigenschaft/ --)

Es liegt eine Änderungsoperation

changeEigenschaft(id, name, wertneu, wertalt) zu einer Eigenschaft vom Typ „Wert“ in nur einer Historie vor.

HK6A = changeEigenschaft(id, name, wert1, wert0) \in EHA \vee

HK6B = changeEigenschaft(id, name, wert1, wert0) \in EHB.

7. Siebter Harmloser Konflikt HK7:

HK7 = (n x changeEigenschaftAdd/ --)

Es liegen n (n ist Element der natürlichen Zahlen und $n > 0$) Einfügeoperationen changeEigenschaftAdd(id, name, eintrag)

- 5 mit dem gleichen Eintrag **eintrag** zur gleichen Eigenschaft vom Typ „aufzählung“ eines Datenelements in einer Historie vor, wobei es in einer anderen Historie keine Löschooperation mit dem gleichen Eintrag zur gleichen Eigenschaft des Datenelements gibt. Die beschriebene Inkonsistenz besteht darin, daß
- 10 in der Kopiedatenbank mit den Erzeugungsoperationen in der der Kopiedatenbank zugeordneten Historie n Einträge der Art **eintrag** mehr vorhanden sind als in der jeweils anderen Kopiedatenbank.

HK7A = n mal changeEigenschaftAdd(id,name,eintrag) \in EHA \vee

- 15 HK7B = n mal changeEigenschaftAdd(id,name,eintrag) \in EHB.

8. Achter Harmloser Konflikt HK8:

HK8 = (n x changeEigenschaftDel/ --)

- Es liegen n (n ist Element der natürlichen Zahlen) Löschooperationen changeEigenschaftDel(id, name, eintrag) mit dem
- 20 gleichen Eintrag **eintrag** zu einer Eigenschaft vom Typ „aufzählung“ eines Datenelements in einer Historie vor, wobei in einer anderen Historie keine Einfügeoperation mit dem gleichen Eintrag zu der Eigenschaft des Datenelements vor-
- 25 liegt. Die beschriebene Inkonsistenz besteht darin, daß in der Kopiedatenbank, in dessen Historie die Löschooperationen stehen, die n Einträge der Art **eintrag** weniger vorhanden sind, als in der jeweils anderen Kopiedatenbank.

HK8A = n mal changeEigenschaftDel(id,name,eintrag) \in EHA \vee

- 30 HK8B = n mal changeEigenschaftDel(id,name,eintrag) \in EHB.

9. Neunter Harmloser Konflikt HK9:

HK9 = (changeEigenschaftInsert/ --)

Es liegt eine Einfügeoperation

- 35 changeEigenschaftInsert(id, name, index1, 1, eintrag1) zu einem Index mit einem einzelnen Eintrag zu einer Eigenschaft vom Typ „geordnete aufzählung“ eines Datenelementes in nur

11. Elfter Harmloser Konflikt HK11:

HK11 = (changeEigenschaftRemove, changeEigenschaftInsert/--)

Es liegt eine Löschoperation

5 changeEigenschaftRemove(id, name, index1, 1, eintrag1) zu einem Index **index** mit einem einzelnen Eintrag zu einer Eigenschaft vom Typ „geordnete aufzählung“ eines Datenelements und eine nachfolgende Erzeugungsoperation

10 changeEigenschaftInsert (id, name, index1, 1, eintrag2) eines Eintrages zum gleichen Index der gleichen Eigenschaft des gleichen Datenelements in nur einer Historie vor.

HK11A = [changeEigenschaftIRemove(id, name, index, 1, eintrag1),
changeEigenschaftIInsert(id, name, index, 1, eintrag2)]
∈ EHA ∨

15 HK11B = [changeEigenschaftIRemove(id, name, index, 1, eintrag1),
changeEigenschaftIInsert(id, name, index, 1, eintrag2)]
∈ EHB.

Im weiteren wird eine Übersicht über kritische Konflikte (KK) d.h. Operationen in mehreren Historien, gegeben:

20

1. Erster Kritischer Konflikt KK1:

KK1 = (createBeziehung12/ createBeziehung13)

Es liegt eine Erzeugung einer Beziehung

25 createBeziehung(beztyp, bname, id1, id2) in einer Historie vor, wobei in einer anderen Historie eine Erzeugung createBeziehung(beztyp, bname, id1, id3) der gleichen Beziehung (beztyp, bname) vom gleichen Quelldatenelement ausgehend aber zu einem anderen Zieldatenelement existiert.

30 KK1 = createBeziehung(beztyp, bname, id1, id2) ∈ EHA ∧
createBeziehung(beztyp, bname, id1, id3) ∈ EHB.

2. Zweiter Kritischer Konflikt KK2:

KK2 = (deleteBeziehung12, createBeziehung13 /
deleteBeziehung12, createBeziehung14)

35 Die unterschiedliche Änderung einer Beziehung kann in den Historien wie der erste kritische Konflikt KK1 erkannt werden. Zusätzlich liegt jedoch in einer gemeinsamen Historie GH eine

Löschung deleteBeziehung(beztyp, bname, id1, id2) der gemeinsamen Beziehung (bname, beztyp) vom gleichen Quelldatenelement, aber zu einem anderen Zieldatenelement vor.

KK2 = deleteBeziehung(beztyp, bname, id1, id2) \in GH \wedge
 5 createBeziehung(beztyp, bname, id1, id3) \in EHA \wedge
 createBeziehung(beztyp, bname, id1, id4) \in EHB.

3. Dritter Kritischer Konflikt KK3:

KK3 = (deleteBeziehung12, createBeziehung13/
 10 deleteBeziehung12)

Die einseitige Veränderung und anderseitige Löschung einer Beziehung kann in den Historien wie der dritte harmlose Konflikt HK3 durch eine Operation

createBeziehung(beztyp, bname, id1, id3) erkannt werden.

15 Zusätzlich liegt jedoch in der gemeinsamen Historie eine Löschung

deleteBeziehung (beztyp, bname, id1, id2) der gemeinsamen Beziehung (bname, beztyp) vom gleichen Quelldatenelement, aber zum letzten gemeinsamen Zieldatenelement vor.

20 KK3A = deleteBeziehung(beztyp, bname, id1, id2) \in GH \wedge
 createBeziehung(beztyp, bname, id1, id3) \in EHA;

KK3B = deleteBeziehung(beztyp, bname, id1, id2) \in GH \wedge
 createBeziehung(beztyp, bname, id1, id3) \in EHB.

25 4. Vierter Kritischer Konflikt KK4:

KK4 = (changeEigenschaft/ changeEigenschaft)

Es liegt eine Änderungsoperation

changeEigenschaft (id, name, wertneu1, wertalt) zu einer Eigenschaft des Typs „wert“ eines Datenelements in einer Historie vor, wobei in einer anderen Historie zur gleichen Eigenschaft des Datenelements eine andere Änderungsoperation existiert.

30 KK4 = changeEigenschaft(id, name, wertneu1, wertalt) \in EHA \wedge
 changeEigenschaft(id, name, wertneu2, wertalt) \in EHB.

einer Historie vor, wobei in einer anderen Historie keine Einfügeoperationen mit einem anderen Eintrag zu nachrechenbarem gleichen Index zu der gleichen Eigenschaft des Datenelementes vorliegt.

5 HK9A = changeEigenschaftInsert(id,name,index,1,eintrag)
 \in EHA \vee

HK9B = changeEigenschaftInsert(id,name,index,1,eintrag)
 \in EHB.

10 10. Zehnter Harmloser Konflikt HK10:

HK10 = (changeEigenschaftRemove/ --)

Es liegt eine Löschoperation

changeEigenschaftRemove(id, name, index1, 1, eintrag1) zu einem Index mit einem einzelnen Eintrag zu einer Eigenschaft

15 vom Typ „geordnete aufzählung“ eines Datenelements in nur einer Historie vor.

HK10A = changeEigenschaftIRemove(id,name,index,1,eintrag)
 \in EHA \vee

20 HK10B = changeEigenschaftIRemove(id,name,index,1,eintrag)
 \in EHB.

5. Fünfter Kritischer Konflikt KK5:

KK5 = (n changeEigenschaftAdd/ m changeEigenschaftDel)

Es liegen n (n ist Element der natürlichen Zahlen) gleiche Operationen der Art

- 5 changeEigenschaftAdd(id, name, eintrag) mit dem gleichen Eintrag zu einer Eigenschaft vom Typ „aufzählung“ eines Datenelements in einer Historie vor, wobei in einer anderen Historie m (m ist Element der natürlichen Zahlen) gleiche Operationen der Art
- 10 changeEigenschaftDel(id, name, eintrag) mit dem gleichen Eintrag zur gleichen Eigenschaft des Datenelements vorliegen. Die beschriebene Inkonsistenz besteht darin, daß in der Kopiedatenbank mit den Erzeugungsoperationen in der der Kopiedatenbank zugeordneten Historie n + m gleiche Einträge eintrag mehr vorhanden sind als in der anderen Kopiedatenbank.
- 15 Um eine exakte Aussage über das Auftreten der Operationen treffen zu können, wird bei dem fünften kritischen Konflikt KK5 zwischen einem fünften kritischen Konflikt erster Art KK5A und einem fünften kritischen Konflikt zweiter Art KK5B unterschieden. Die Zuordnung erfolgt dabei über die Erzeugungsoperationen.
- 20

KK5A = (n changeEigenschaftAdd(id,name,eintrag) \in EHA \wedge
m changeEigenschaftDel(id,name,eintrag) \in EHB \vee

25 KK5B = (m changeEigenschaftDel(id,name,eintrag) \in EHA;
n changeEigenschaftAdd(id,name,eintrag) \in EHB.

6. Sechster Kritischer Konflikt KK6:

KK6 = (changeEigenschaftInsert / changeEigenschaftInsert)

Es liegt eine Einfügeoperation

changeEigenschaftInsert(id, name, index1, 1, eintrag1) zu einem Index mit einem einzelnen Eintrag zu einer Eigenschaft vom Typ „geordnete aufzählung“ eines Datenelements in einer Historie vor, wobei in einer anderen Historie eine Einfügeoperation mit einem anderen Eintrag zum nachrechenbar gleichen Index der Eigenschaft des Datenelements vorliegt.

10 KK6 = changeEigenschaftInsert(id, name, index1, 1, eintrag1)
 \in EHA \wedge
 changeEigenschaftInsert(id, name, index2, 1, eintrag2)
 \in EHB.

15 7. Siebter Kritischer Konflikt KK7:

KK7 = (changeEigRemove, changeEigInsert /
 changeEigRemove, changeEigInsert)

Die beidseitig unterschiedliche Änderung eines einzelnen Eintrages an einem gemeinsamen Index **index** zu einer Eigenschaft vom Typ „geordnete aufzählung“ eines Datenelementes ist in den Historien wie der sechste kritische Konflikt KK6 erkennbar. Zusätzlich liegt jedoch in der gemeinsamen Historie eine changeEigenschaftRemove(id, name, index1, 1, eintrag1)-Operation des letzten gemeinsamen Eintrages am nachrechenbar gleichen Index vor.

25 KK7 = changeEigenschaftRemove(id, name, index1, 1, eintrag1)
 \in GH \wedge
 changeEigenschaftInsert(id, name, index2, 1, eintrag2)
 \in EHA \wedge
 30 changeEigenschaftInsert(id, name, index3, 1, eintrag3)
 \in EHB.

8. Achter Kritischer Konflikt KK8:

KK8 = (changeEigRemove, changeEigInsert / changeEigRemove)

Die einseitige Änderung und anderseitige Löschung eines einzelnen Eintrages an einem gemeinsamen Index **index** zu einer

5 Eigenschaft vom Typ „geordnete aufzählung“ eines Datenelements in den Historien ist wie ein zehnter harmloser Konflikt HK10 erkennbar. Zusätzlich liegt jedoch in der gemeinsamen Historie eine

changeEigenschaftRemove(id, name, index1, 1, eintrag1)-

10 Operation des letzten gemeinsamen Eintrages am nachrechenbar gleichen Index vor.

KK8A = changeEigenschaftRemove(id, name, index1, 1, eintrag1)

∈ GH ∧

changeEigenschaftInsert(id, name, index2, 1, eintrag2)

15 ∈ EHA;

KK8B = changeEigenschaftRemove(id, name, index1, 1, eintrag1)

∈ GH ∧

changeEigenschaftInsert(id, name, index2, 1, eintrag2)

∈ EHB.

20

Die in den Historien gespeicherten Operationen beschreiben die autonom veränderten Datenbereiche direkt und werden zur Beschreibung sowie zur im weiteren beschriebenen Behebung der Inkonsistenzen verwendet.

25

Zur Erkennung der Inkonsistenzen werden jeweils zwei Historien miteinander verglichen.

Die Erkennung der Inkonsistenzen erfolgt zu Beginn des Verfahrens, vor der eigentlichen Reintegration.

30

Die Suche nach den vorhandenen Inkonsistenzen in den Kopiedatenbanken durch die Suche nach Konfliktoperationen erfolgt in den im weiteren beschriebenen drei Schritten.

35

- In einem ersten Schritt werden die zwei miteinander zu vergleichenden Historien von den Kopiedatenbanken bzw. der Da-

tenbank durchlaufen, die miteinander abgeglichen werden sollen. Alle Operationen der Historien werden auf jeder Seite getrennt jeweils einer der oben beschriebenen neuen Operationsmengen (createElement-Operationen, deleteElement-Operationen, createBeziehung-Operationen, delete-Beziehung-Operationen, changeEigenschaft-Operationen, changeEigenschaftAdd-Operationen, changeEigenschaftDel-Operationen, changeEigenschaftInsert-Operationen und ChangeEigenschaftRemove-Operationen) zugeordnet.

- In einem zweiten Schritt wird für jeden oben beschriebenen Konflikttyp HK1A, ..., HK11A, HK1B, ..., HK11B, KK1, KK2, KK3A, KK3B, KK4, KK5A, KK5B, KK6, KK7, KK8A, KK8B jeweils ein Konfliktregister KR angelegt. Dabei wird gewährleistet, daß alle Konflikte, bei denen Operationen aus beiden Historien zum jeweiligen Konflikt beitragen, nicht doppelt erkannt und doppelt in dem jeweiligen Konfliktregister KR abgelegt werden. Danach werden entsprechend den Definitionen der Konflikttypen, wie oben beschrieben, beginnend mit dem ersten harmlosen Konflikt HK1A in der Historie des ersten Benutzers A, die gerade gebildeten Operationsmengen durchsucht. Wurde ein Konflikt ermittelt, so wird der Konflikt im Konfliktregister KR des entsprechenden Konflikttyps abgelegt, beispielsweise wird der erste harmlose Konflikt HK1 in der Kopiedatenbank des ersten Benutzers A in einem Konfliktregister KR_HK1A abgelegt.

- Ist die Suche und Speicherung der Konflikte erfolgt, werden in einem dritten Schritt die zu Beginn erstellten Operationsmengen wieder gelöscht. Die nachfolgende Behebung der Inkonsistenzen beruht auf den in den Konfliktregistern KR abgelegten Konflikten und deren Operationen.

Ein Konfliktregister KR ist wie folgt definiert:

Konfliktregister KR:

Ein Konfliktregister KR zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 2-Tupel

5 $KR(EH_A, EH_B, GH) \underline{def} (krtyp, konfliktids)$

• $krtyp \in \{KA_HK1A, \dots, KA_HK11A, KA_HK1B, \dots, KA_HK11B, KA_KK1, KA_KK2, KA_KK3A, KA_KK3B, KA_KK4, KA_KK5A, KA_KK5B, KA_KK6, KA_KK7, KA_KK8A, KA_KK8B\}$

10 • konfliktids sind Identifier aller dem Konfliktregister KR zugeordneten Konflikte $K(EHA, EHB, GH)$, wobei $K.typ$ dem jeweiligen Konfliktarraytyp $KR.krtyp$ zuzuordnen ist.

15 Eine Anomalie liegt dann vor, wenn zwei Datenelemente in beiden Kopiedatenbanken vor und nach der Teilung existieren und diese nach der Teilung durch eine gerichtete Beziehung vom gleichen Typ $beztyp$ und mit gleichem Namen $bname$, jedoch mit vertauschtem Quelldatenelement und Zieldatenelement verbunden sind. Während der Reintegration muß mindestens eine dieser
20 Beziehungen abgelehnt oder müssen beide verändert werden.

Unter einer gerichteten Beziehung ist eine Beziehung zu verstehen, die von einem Zieldatenelement zu einem Quelldatenelement gerichtet ist.

25

Eine Anomalie ist wie folgt definiert:

Anomalie:

Eine Anomalie AM zweier Konflikte K1 (EHA, EHB, GH) und K2 (EHA, EHB, GH) ist ein 4-Tupel

5 AM(K1, K2) def (id, amtyp, kid1, kid2)

- id ist ein systemweit eindeutiger Identifier (siehe auch Definition eines Datenelementes)
- amtyp \in {Anomalie1A, ..., Anomalie16AB}
- 10 • kid1 = K1.id
- kid2 = K2.id.

Ein Anomalieregister AMR zweier Historien ist wie folgt definiert:

15

Anomalieregister AMR:

Ein Anomalieregister AMR zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 1-Tupel

20 AMR (EHA, EHB, GH) def (anomalieids)

- anomalieids sind die Identifier aller Anomalien der Historien EHA und EHB und der gemeinsamen Historie GH.

25 Pseudo-Anomalien beschreiben Situationen, in denen das Entstehen einer Anomalie aus vorliegenden Konflikten nur durch eine gezielte Minimierung der Entscheidungsmöglichkeiten der Konflikte vermieden werden kann.

30 Eine Pseudo-Anomalie ist, wie im folgenden dargestellt, definiert:

Pseudo-Anomalie PAM:

Eine Pseudo-Anomalie PAM zweier Konflikte $K1(EHA, EHB, GH)$ und $K2(EHA, EHB, GH)$ ist ein 4-Tupel

5 $PAM(K1, K2) \underline{def} (id, pamtyp, kid1, kid2)$

- id ist ein systemweit eindeutiger Identifier (siehe auch Definition eines Datenelementes)
- $pamtyp \in \{Pseudo-Anomalie1A, \dots, Pseudo-Anomalie32AB\}$
- 10 • $kid1 = K1.id$
- $kid2 = K2.id$.

Ein Pseudo-Anomalieregister PAMR ist wie folgt definiert:

15 **Pseudo-Anomalieregister PAMR:**

Ein Pseudo-Anomalieregister PAMR zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 1-Tupel

$PAMR(EHA, EHB, GH) \underline{def} (Pseudo-Anomalieids)$

20

- Pseudo-Anomalieids sind die Identifier aller Pseudo-Anomalien der Historien EHA und EHB sowie der gemeinsamen Historie GH.

25 Nach Ermittlung der Konflikte wird jeder ermittelte Konflikt jeweils durch eine einzelne Entscheidung gelöst. Der Konfliktlösungsprozeß besteht somit aus einer Sequenz von Konfliktlösungsentscheidungen.

30 Die Konfliktlösung ist in Fig.1 mit Schritt 104 bezeichnet.

Grundsätzlich gibt es verschiedene Entscheidungsmöglichkeiten:

- a) Annahme der Konfliktoperation(en)
- 35 b) Ablehnung der Konfliktoperation(en)
- c) Teilweise Annahme, teilweise Ablehnung der Konfliktoperation(en)

d) Ablehnung der Konfliktoperation(en), Annahme neuer erzeugter Operation(en).

5 Den einzelnen Konflikttypen wird ein Entscheidungsset ES zugeordnet, wobei der Entscheidungsset ES mögliche Entscheidungen enthält, mit denen eine durch eine Operation des jeweiligen Konflikttyps, dem jeweils ein Entscheidungsset ES zugeordnet ist, erzeugte Inkonsistenz behoben werden kann.

10 In der Fig.4 ist eine Zusammenstellung aller Entscheidungssets, die jeweils einem Konflikt zugeordnet sind, dargestellt.

15 Es wird jeweils in einer Zeile der in Fig.4 dargestellten Tabelle eine mögliche Entscheidungsmöglichkeit E1, E2, E3a, E3b, E4, E5a, E5b, E6 dargestellt.

20 Mit einem x in einem Feld ist jeweils bezeichnet, daß der jeweils in der Spalte aufgeführte Konflikt durch eine Entscheidungsmöglichkeit, die in der jeweiligen Zeile dargestellt ist, gelöst werden kann.

Im weiteren wird eine Übersicht über die möglichen Entscheidungsmöglichkeiten gegeben:

25

Eine erste Entscheidungsmöglichkeit E1 beschreibt die Annahme einer Konfliktoperation oder mehrerer Konfliktoperationen.

30 Eine Konfliktoperation beschreibt alle zu einem Konflikt gehörenden Datenoperationen. Unter Annahme wird verstanden, daß die Konfliktoperationen in der Kopiedatenbank, in der sie noch nicht vorgenommen worden sind, ausgeführt werden.

35 Eine zweite Entscheidungsmöglichkeit E2 beschreibt die Ablehnung einer Konfliktoperation oder mehrerer Konfliktoperationen.

Eine dritte Entscheidungsmöglichkeit E3 beschreibt die Annahme einer oder mehrerer Konfliktoperationen(en) in einer Kopiedatenbank und die Ablehnung der Konfliktoperation(en) in der anderen Kopiedatenbank.

5

Für die dritte Entscheidungsmöglichkeit E3 ist eine Detailentscheidung vorgesehen, die definiert, welche der in den Historien verschiedener Kopiedatenbanken der Benutzer A und B vorliegenden Konfliktoperationen angenommen und welche abgelehnt werden sollen.

10

Diese Entscheidungsmöglichkeiten werden als erster Teil E3a der dritten Entscheidungsmöglichkeit E3 und zweiter Teil E3b der dritten Entscheidungsmöglichkeit E3 bezeichnet. Der erste Teil E3a der dritten Entscheidungsmöglichkeit E3 beschreibt die Annahme der Konfliktoperation(en) in der Kopiedatenbank des ersten Benutzers A und die Ablehnung der Konfliktoperation(en) in der Kopiedatenbank des zweiten Benutzers B. Der zweite Teil E3b der dritten Entscheidungsmöglichkeit E3 beschreibt die Annahme der Konfliktoperation(en) der Kopiedatenbank des zweiten Benutzers B und die Ablehnung der Konfliktoperation(en) der Kopiedatenbank des ersten Benutzers A.

20

Wie in Fig.4 dargestellt, beschreibt ein erster Entscheidungsset ES1, der dem ersten harmlosen Konflikt HK1 zugeordnet ist, die erste Entscheidungsmöglichkeit E1 sowie die zweite Entscheidungsmöglichkeit E2 zur Erhebung des ersten harmlosen Konflikts HK1.

25

Ein zweiter Entscheidungsset ES2 ist dem zweiten harmlosen Konflikt HK2 zugeordnet und enthält wiederum die erste Entscheidungsmöglichkeit E1 sowie die zweite Entscheidungsmöglichkeit E2 zur Behebung des zweiten harmlosen Konflikts HK2.

30

Entsprechen die bisherigen Lösungsmöglichkeiten durch ein Annehmen oder Ablehnen vorhandener Konfliktoperationen nicht den Zielvorstellungen der Benutzer bezüglich der letztendli-

35

chen reintegrierten Datenbank, d.h. können die Benutzer A und B sich nicht auf eine durch die Konflikte beschriebenen Zustände einigen, so gibt es die Möglichkeit der Annahme einer Zwischenlösung oder die Möglichkeit der Auswahl und Annahme neuer, nicht im Entscheidungsset enthaltener Operationen. Beide Möglichkeiten sollen nachfolgend dargestellt werden.

Für einen Konflikt mit einer Anzahl n den Konflikt definierenden, gleichen Operationen aus einem Set an Datenoperationen in nur einer Historie (Konflikte der Typen HK7 II und HK8 II) gibt es generell im weiteren beschriebene Auswahlmöglichkeiten von Zwischenzuständen.

Für den siebten harmlosen Konflikt HK7 mit $n = 1$ wird nachfolgend HK7 I und für den siebten harmlosen Konflikt HK7 mit $n > 1$ wird nachfolgend HK7 II geschrieben. Für den achten harmlosen Konflikt HK1 mit $m = 1$ wird nachfolgend HK8 I und für den achten harmlosen Konflikt HK8 mit $n > 1$ wird nachfolgend HK8 II geschrieben.

Eine vierte Entscheidungsmöglichkeit E4 beschreibt dabei eine teilweise Annahme und teilweise Ablehnung der Konfliktoperationen.

Für die vierte Entscheidungsmöglichkeit E4 ist eine Präzisierung vorgesehen, die definiert, wieviele der einseitig auftretenden Konfliktoperationen angenommen und wieviele abgelehnt werden sollen. Die Entscheidungsmöglichkeiten reichen bei einer Anzahl von n Operationen (n ist Element der natürlichen Zahlen), von einer Annahme einer Operation und einer Ablehnung von $n-1$ Operationen bis zu einer Annahme von $n-1$ und einer Ablehnung von einer Operation. Als Entscheidungsmöglichkeit reicht dabei aus, die Anzahl k ($0 < k < n$) der angenommenen Operationen zu definieren. Die Anzahl der abgelehnten Operationen errechnet sich dann aus $n - k$. Die vierte Entscheidungsmöglichkeit E4 kann somit folgendermaßen spezia-

lisiert werden: Die vierte Entscheidungsmöglichkeit E4 beschreibt die Anzahl der angenommenen Konfliktoperationen k .

Im weiteren wird der fünfte kritische Konflikt KK5 mit
5 $n = m = 1$ nachfolgend mit KK5 I und für den fünften kritischen Konflikt KK5 mit $n > 1$ und $m > 1$ nachfolgend KK5 II bezeichnet.

Für einen fünften kritischen Konflikt KK5 II mit einer Anzahl
10 n den Konflikt definierenden, gleichen Operationen `changeEigenschaftAdd` bei einer Kopiedatenbank und einer Anzahl m den Konflikt definierenden gleichen Operationen `changeEigenschaftDel` in einer anderen Kopiedatenbank gestaltet sich die Auswahl der Zwischenzustände schwieriger.

15 Da die den Konflikt definierenden Operationen in der einen Historie und die am Konflikt beteiligten Operationen in der anderen Historie einander auslöschend sind, kann das gleiche Endergebnis durch verschiedene Entscheidungsmöglichkeiten getroffen werden.
20

So kann die Rücksetzung einer `changeEigenschaftAdd`-Operation und die Annahme einer `changeEigenschaftAdd`-Operation in der einen Kopiedatenbank verbunden mit der Rücksetzung einer
25 `changeEigenschaftDel`-Operation bei einer anderen Kopiedatenbank das gleiche Ergebnis erzielen, wie die Annahme zweier `changeEigenschaftAdd`-Operationen der einen Kopiedatenbank und einer `changeEigenschaftDel`-Operation bei der anderen Kopiedatenbank. Damit alle Entscheidungsmöglichkeiten zwischen den
30 Extrema Ablehnung der Operationen der einen Kopiedatenbank und Annahme der Operationen bei der anderen Kopiedatenbank (dritte Entscheidungsmöglichkeit E3a/E3b) gegeben sind, gleichzeitig aber vermieden wird, daß verschiedene Entscheidungsmöglichkeiten ein gleiches Ergebnis liefern, gibt es die
35 unten aufgeführte Lösungsmöglichkeit der fünften Entscheidungsmöglichkeit E5.

Die fünfte Entscheidungsmöglichkeit E5 kann durch Ablehnung aller Operationen den letzten gemeinsamen Stand zwischen den Kopiedatenbanken erzeugen und ermöglicht alle anderen Endzustände aus den Kombinationen der Operationen, jedoch ohne die
5 oben vorliegenden Konfliktzustände.

Die fünfte Entscheidungsmöglichkeit E5 beschreibt also die teilweise Annahme und teilweise Ablehnung der Konfliktoperationen bei einer Kopiedatenbank und die Ablehnung der Konfliktoperationen bei einer anderen Kopiedatenbank.
10

Für die fünfte Entscheidungsmöglichkeit E5 sind die Teilentscheidungsmöglichkeiten notwendig, die einerseits definieren, welche Kopiedatenbank von der teilweisen Annahme und teilweisen Ablehnung und welche Datenbank von der vollständigen Ablehnung betroffen ist und andererseits, wieviele der Konfliktoperationen bei einer teilweisen Annahme angenommen und wieviele abgelehnt werden sollen. Die Definition der Anzahl an Operationen bei teilweiser Annahme und teilweiser Ablehnung
15 kann dabei wie bei der vierten Entscheidungsmöglichkeit E4 über die alleinige Definition der Anzahl der angenommenen Operationen erfolgen. Die Anzahl der angenommenen Operationen in einer ersten Kopiedatenbank des ersten Benutzers A werden dabei mit i und die Anzahl der angenommenen Operationen der
20 Kopiedatenbank des zweiten Benutzers B mit k bezeichnet. Somit sind Detailentscheidungen der fünften Entscheidungsmöglichkeit E5:

Erster Teil E5a der fünften Entscheidungsmöglichkeit E5:
30 Anzahl der angenommenen Konfliktoperationen der Kopiedatenbank des ersten Benutzers A:
($1 < i < n$), wenn es sich um eine changeEigAdd-Operation handelt,
($1 < k < m$), wenn es sich um eine changeEigDel-Operation
35 handelt, und Ablehnung aller Konfliktoperationen der Kopiedatenbank des zweiten Benutzers B.

Zweiter Teil E5b der fünften Entscheidungsmöglichkeit E5:
Anzahl der angenommenen Konfliktoperationen der Kopiedatenbank des zweiten Benutzers B:

($1 < i < n$), wenn es sich um eine changeEigAdd-Operation
5 handelt,

($1 < k < m$), wenn es sich um eine changeEigDel-Operation
handelt und Ablehnung aller Konfliktoperationen der Kopiedatenbank des ersten Benutzers A.

- 10 Eine Auswahlmöglichkeit für die Erzeugung eines neuen Zustandes bzgl. des Konfliktes wird durch die nachfolgenden Entscheidungsmöglichkeiten definiert werden.

15 Generell ist die Möglichkeit der Erzeugung und Auswahl eines sich von den beiden vorliegenden Versionen unterscheidenden Zustandes für alle Konflikte außer den Konflikten des Typs HK1, HK2, HK4, HK10 gegeben.

20 Für eine Erzeugung eines neuen Zustandes bedarf es der Schaffung einer gemeinsamen Ausgangsposition, d.h. die betreffende(n) Operation(en) müssen abgelehnt und beide Kopiedatenbanken bezüglich der von dem Konflikt betroffenen Datenstruktur konsistent gemacht werden. Diese Ablehnung der Operationen ist bei sich überschreibenden Operationen des Typs
25 changeEigenschaft() (HK6, HK4) nicht erforderlich, da die mit dem neuen Zustand erzeugte Operation die alten Operationen direkt überschreibt.

30 Für einen Konflikt mit einer den Konflikt definierenden Operation aus dem Set an Datenoperationen (Konflikte der Typen HK3, HK6, HK7 I, HK8 I und HK9), für einen Konflikt mit mehreren den Konflikt definierenden Operationen aus dem Set an Datenoperationen bei nur einer Kopiedatenbank (Konflikte der Typen HK5, HK7 II, HK8 II, HK11) sowie für einen Konflikt mit
35 mindestens einer den Konflikt definierenden Operation aus dem Set an Datenoperationen bei beiden Kopiedatenbanken (Konflikte der Typen KK1, ..., KK8) gibt es folgende Lösungs-

möglichkeit, die als sechste Entscheidungsmöglichkeit E6 bezeichnet wird.

Die sechste Entscheidungsmöglichkeit E6 beschreibt die Ablehnung der Konfliktoperation(en) und Auswahl neuer Operation(en).

Für die beidseitige Änderung einer Beziehung (KK2) oder die beidseitige Änderung eines Eintrages in einer geordneten Aufzählung (KK7) bezieht sich die Ablehnung dabei, im Gegensatz zur zweiten Entscheidungsmöglichkeit E2 (Ablehnung aller Operationen) nur auf die Erzeugeroperationen der neuen Beziehungen bzw. der neuen Einträge. Die gemeinsame Löschoption der alten Beziehung oder des alten Eintrages bleibt unberührt. Für den dritten kritischen Konflikt KK3 und den achten kritischen Konflikt KK8 gibt es bei der sechsten Entscheidungsmöglichkeit E6 nur die Möglichkeit, die createBeziehung-Operation oder die changeEigenschaftInsert-Operation zu ändern. Die gemeinsame Löschoption bleibt unberührt.

Für die Definition eines neuen Zustandes eines Konflikts vom Typ HK7, HK8 oder KK5 werden die Anzahl der changeEigenschaftAdd-Operationen und changeEigenschaftDel-Operationen ebenfalls mit i und k beschrieben (i, wenn es sich um changeEigAdd-Operationen handelt und k, wenn es sich um changeEigDel-Operationen handelt).

Für die Auswahl eines neuen Zustandes und die Erzeugung der dafür notwendigen Operationen sind Interaktionen auf der Oberfläche des Anwendungsprogramms üblich. Besteht die Möglichkeit einer Auswahl eines neuen Zustandes nur über eine komplexe Interaktion und betreffen die durch die Interaktion erzeugten Operationen auch noch andere, nicht gelöste Konflikte, so besteht die Möglichkeit

- a) mit dem Entscheidungsset die anderen betroffenen Konflikte ebenfalls zu lösen oder
b) die Erzeugung des neuen Zustandes auf einen späteren Zeitpunkt, d.h. nach der Reintegration und während der gekoppelten Weiterarbeit zu verschieben.

Sollen nach a) andere betroffene Konflikte ebenfalls gelöst werden, so ist vorher die Ablehnung der Operationen des Konfliktsets notwendig, wenn es sich um Konflikte vom Typ HK6 und KK4 handelt und die Operationen keine sich überschreibenden Operationen sind.

Fig.4 zeigt die Entscheidungssets ES1, ES2, ES3, ES4, ES5, ES6, ES7, ES8, ES9, ES10, ES11, ES12, ES13, ES14, ES15, ES16, ES17, ES18, ES19, ES20, ES21, ES22, die den jeweiligen Konflikten zugeordnet sind.

Dem sechsten harmlosen Konflikt HK6 ist ein sechster Entscheidungsset ES6 zugeordnet, der die erste Entscheidungsmöglichkeit E1, die zweite Entscheidungsmöglichkeit E2 sowie die sechste Entscheidungsmöglichkeit E6 enthält.

Ein zwölftes Entscheidungsset ES12 ist dem ersten kritischen Konflikt KK1 zugeordnet. Der zwölfte Entscheidungsset ES12 enthält vier mögliche Entscheidungen, die zweite Entscheidungsmöglichkeit E2, den ersten Teil E3a der dritten Entscheidungsmöglichkeit E3, den zweiten Teil E3b der dritten Entscheidungsmöglichkeit E3 sowie die sechste Entscheidungsmöglichkeit E6.

Die weiteren Entscheidungssets sind in Fig.4 dargestellt und sollen zur Vereinfachung im folgenden anhand der folgenden Liste dargestellt werden:

- Dem ersten harmlosen Konflikt HK1, dem zweiten harmlosen Konflikt HK2, dem vierten harmlosen Konflikt HK4 sowie dem zehnten harmlosen Konflikt HK10 sind jeweils Entscheidungs-

sets zugeordnet, die die erste Entscheidungsmöglichkeit E1 sowie die zweite Entscheidungsmöglichkeit E2 umfassen.

- Dem dritten harmlosen Konflikt HK3, dem fünften harmlosen Konflikt HK5, dem sechsten harmlosen Konflikt HK6, dem ersten Typ HK7 I des siebten harmlosen Konflikts HK7, dem ersten Typ HK8 I des achten harmlosen Konflikts HK8, dem neunten harmlosen Konflikt HK9 sowie dem elften harmlosen Konflikt HK11 sind jeweils Entscheidungssets zugeordnet, die die erste Entscheidungsmöglichkeit E1, die zweite Entscheidungsmöglichkeit E2 sowie die sechste Entscheidungsmöglichkeit E6 enthalten.
- Dem zweiten Typ HK7 II des siebten harmlosen Konflikts HK7, dem zweiten Typ HK8 II des achten harmlosen Konflikts HK8, dem ersten kritischen Konflikt KK1, dem zweiten kritischen Konflikt KK2, dem dritten kritischen Konflikt KK3, dem vierten kritischen Konflikt KK4, dem ersten Typ KK5 I des fünften kritischen Konflikts KK5, dem sechsten kritischen Konflikt KK6, dem siebten kritischen Konflikt KK7 und dem kritischen Konflikt KK8 sind jeweils Entscheidungssets zugeordnet, die die zweite Entscheidungsmöglichkeit E2, den ersten Teil E3a der dritten Entscheidungsmöglichkeit E3, den zweiten Teil E3b der dritten Entscheidungsmöglichkeit E3 sowie die sechste Entscheidungsmöglichkeit E6 enthalten.
- Dem zweiten Typen KK5 II des fünften kritischen Konflikts KK5 ist ein Entscheidungsset mit sechs möglichen Entscheidungen, der zweiten Entscheidungsmöglichkeit E2, dem ersten Teil E3a der dritten Entscheidungsmöglichkeit E3, dem zweiten Teil E3b der dritten Entscheidungsmöglichkeit E3, dem ersten Teil E5a der fünften Entscheidungsmöglichkeit E5, dem zweiten Teil E5b der fünften Entscheidungsmöglichkeit E5 sowie der sechsten Entscheidungsmöglichkeit E6, zugeordnet.
- Dem zweiten Typ HK7 II des siebten harmlosen Konflikts HK7 und dem zweiten Typ HK8 II des achten harmlosen Konflikts HK8 ist jeweils ein Entscheidungsset mit vier möglichen Entscheidungen, der ersten Entscheidungsmöglichkeit E1, der zweiten Entscheidungsmöglichkeit E2, der vierten Entschei-

dungsmöglichkeit E4 sowie der sechsten Entscheidungsmöglichkeit E6 zugeordnet.

Einschränkungen von Entscheidungsmöglichkeiten

5

Es ist zu bemerken, daß die korrekte Ausführung einzelner Entscheidungen zum Vorhandensein eines Datenelements oder mehrerer Datenelemente in beiden Kopiedatenbanken abhängig ist.

10

Beispielsweise müssen für eine Annahme bei dem dritten harmlosen Konflikt HK3A beide in der Beziehungsoperation in Konflikt über ihre Identifizier bezeichneten Datenelemente auch in der Kopiedatenbank des Benutzers B vorhanden sein. Fehlt eines der Datenelemente oder fehlen gar beide, so ist diese Entscheidung über eine Annahme der Operation nicht möglich.

15

Somit ist erkennbar, daß zwischen einzelnen Konflikten Abhängigkeiten bestehen können.

20

Eine Abhängigkeit eines Konflikts zu Konflikten des Typs HK1A, HK1B, HK2A oder HK2B bezüglich seiner Entscheidungsmöglichkeiten ist wie folgt definiert:

25 Abhängiger Konflikt:

Ein Konflikt ist von dem ersten harmlosen Konflikt HK1 oder dem zweiten harmlosen Konflikt HK2 dann abhängig, wenn seine Entscheidungsmöglichkeiten durch das Vorhandensein eines ersten harmlosen Konflikts HK1 oder eines zweiten harmlosen

30

Konflikts HK2 eingeschränkt werden.

Eine Abhängigkeit AK eines Konflikts ist wie folgt definiert:

Abhängigkeit AK eines Konflikts:

Eine Abhängigkeit AK eines Konfliktes $K1(EHA, EHB, GH)$ von einem Konflikt $K1(EHA, EHB, GH)$ ist ein 4-Tupel

5 $AK(K1, K2) \underline{def} (id, ktyp, kid1, kid2)$

- id ist ein systemweit eindeutiger Identifier (siehe auch Definition eines Datenelementes)
- $ktyp \in \{HK1A, HK1B, HK2A, HK2B\}$
- 10 • $kid1 = K1.id$
- $kid2 = K2.id$.

Ein Abhängigkeitsregister AKR ist wie folgt definiert:

15 **Abhängigkeitsregister AKR:**

Ein Abhängigkeitsregister AKR zweier Historien EHA und EHB und einer gemeinsamen Historie GH ist ein 1-Tupel

$AKR(EH_A, EH_B, GH) \underline{def} (abhängigkeitsids)$

20

- $abhängigkeitsids$ sind die Identifier aller erkannten Abhängigkeiten der Historien EH_A und EH_B und der gemeinsamen Historie GH.

25 Alle Einschränkungen von Entscheidungsmöglichkeiten durch vorhandene Konflikte des ersten harmlosen Konflikts HK1 und des zweiten harmlosen Konflikts HK2 werden zu Beginn der Konfliktlösung auf der Grundlage der Abhängigkeiten AK erkannt und markiert.

30

Dazu dient der in der Konfliktdefinition für jeden Konflikt eingeführte Parameter der Entscheidungseinschränkungen. Nachfolgend werden alle möglichen Einschränkungen beschrieben.

35 Der erste harmlose Konflikt HK1 beeinträchtigt die Entscheidungsmöglichkeiten von Konflikten mit Operationen innerhalb der eigenen Historie. Zu diesen abhängigen Konflikten gehören

alle, die eine createBeziehung-Operation oder eine Eigenschaftsänderung mit dem erzeugten Datenelement enthalten. Dabei werden bei harmlosen Konflikten HK6, ..., HK11 mit Eigenschaftsoperationen zu diesem Datenelement die Entscheidungen
5 um eine Annahme und eine Erzeugung eines neuen Zustandes minimiert.

Die Entscheidungsmöglichkeiten der kritischen Konflikte KK1, KK2 und KK3 mit Beziehungsoperationen mit dem erzeugten Datenelement werden um die Möglichkeit der Annahme der Operationen (-E3a, -E3b) der Kopiedatenbank reduziert, in dessen
10 Historie die createElement-Operation steht. Die Entscheidungsmöglichkeiten für den dritten harmlosen Konflikt HK3 und den fünften harmlosen Konflikt HK5 mit Beziehungsoperationen
15 mit dem erzeugten Datenelement werden um die Entscheidung der Übernahme und/oder der Erzeugung eines neuen Zustandes reduziert. Kritische Konflikte mit Eigenschaftsoperationen erfahren keine Veränderungen ihrer Entscheidungsmenge
(Entscheidungssets).

20

Ein zweiter harmloser Konflikt HK2 beeinträchtigt die Entscheidungsmöglichkeit von Konflikten beider Kopiedatenbanken. Die harmlosen Konflikte zur Veränderung von Eigenschaften HK6, ..., HK11 werden wie bei dem ersten harmlosen Konflikt
25 HK1 behandelt. Die harmlosen Konflikte mit Beziehungsoperationen (HK4, HK5) in der Historie der deleteElement-Operation besitzen keine Entscheidungsmöglichkeit zur Ablehnung mehr und die Entscheidungen zur harmlosen Beziehungskonflikten der anderen Kopiedatenbank HK3, HK5 werden um die Möglichkeit der
30 Übernahme und/oder Erzeugung eines neuen Zustandes minimiert. Die kritischen Konflikte mit Beziehungsoperationen der Kopiedatenbank ohne die deleteElement-Operation werden um die Möglichkeit des Rücksetzens und/oder der Annahme der Operationen reduziert. Kritische Konflikte mit Eigenschaftsoperationen
35 erfahren auch hier keine Veränderung ihrer Entscheidungsmenge.

Eine gemeinsame deleteElement-Operation, die in der gemeinsamen Historie GH enthalten ist und keinen Konflikt darstellt, verringert in speziellen Fällen die Entscheidungsmöglichkeiten. So können alle kritischen Konflikte KK2 und KK3 mit Beziehungsooperationen, in denen das Zieldatenelement der gemeinsamen deleteBeziehung-Operation dem beidseitig gelöschten Datenelement entspricht, nicht mehr rückgesetzt werden.

Abhängige Konflikte mit einer oder mehreren Beziehungsooperationen (HK3, HK4, HK5, KK1, KK2, KK3), d.h. mit mehreren verzeichneten Identifiern und damit mehreren beteiligten Datenelementen, können gleichzeitig mehrere Abhängigkeiten aufweisen. Dabei kann es pro auftretenden Identifier in einem Konflikt nur eine Abhängigkeit geben. Ein abhängiger Konflikt mit mehreren Identifiern kann einerseits mehrere Abhängigkeiten zu Konflikten desselben Konflikttyps (zum Beispiel zu zwei Konflikten des ersten harmlosen Konflikts HK1a) und andererseits mehrerer Abhängigkeiten zu Konflikten unterschiedlicher Konflikttypen (z.B. zu einem Konflikt vom Typ HK1a und zu einem Konflikt vom Typ HK1b) aufweisen. Beispielsweise kann ein abhängiger Konflikt vom Typ HK3a zwei Abhängigkeiten zu zwei Konflikten des Typs HK1a haben, nämlich eine mit dem Identifier id1 und eine mit dem Identifier id2. Der zweite kritische Konflikt KK2 kann gleichzeitig eine Abhängigkeit zu dem zweiten harmlosen Konflikt HK2a, zu dem ersten harmlosen Konflikt HK1a und zu dem ersten harmlosen Konflikt HK1B aufweisen.

Durch die Abhängigkeit der Identifier zu einem Konflikt vom Typ HK1 oder HK2 kann es maximal pro Konflikt vom Typ HK5, KK1, KK2 oder KK3 drei Einschränkungen geben. Alle Konflikte vom Typ HK3A, HK3B, HK4A, HK4B, HK5A und HK5B können gleichzeitig mehrere Abhängigkeiten zu Konflikten desselben oder unterschiedlichen Typs haben. Konflikte der Typen KK1, KK2, KK3A und KK3B können dagegen gleichzeitig mehrere Abhängigkeiten zu Konflikten verschiedener Typen aufweisen. Es können jedoch bei mehrfach vom selben Konflikttyp abhängigen Kon-

flikten auch die gleichen Einschränkungen der Entscheidungsmöglichkeiten mehrfach auftreten. So sind für einen Konflikt vom Typ HK3A gleichzeitig zwei Abhängigkeiten bei Vorhandensein eines Konflikts vom Typ HK1A oder eines vom Typ HK2B
5 möglich und die Entscheidungen werden jeweils um die erste Entscheidung E1 minimiert.

Die sich zu manchen abhängigen Konflikten ergebenden mehrfachen Einschränkungen mit den gleichen Entscheidungsmöglichkeiten bedürfen keiner gesonderten Betrachtung. Jede dieser
10 Entscheidungseinschränkungen wird betrachtet, als ob es eine eigene, spezielle Einschränkung ist. Es werden somit alle mehrfachen Einschränkungen, wie andere auch, im Konflikt vermerkt. Bei einer Lösung eines Konflikts des Typs HK1 und HK2
15 in der Art, daß eine der mehrfach vorkommenden Einschränkungen aufgehoben wird, bleiben die restlichen dieser Einschränkungen erhalten. Erst wenn durch verschiedene Konfliktlösungen keine der mehrfachen Einschränkungen mehr vorhanden ist, kann eine Entscheidung dieser Art vorgenommen werden. Dies
20 gilt unabhängig davon, ob die Einschränkung einmal oder mehrmals vorhanden war.

Die einmalig zu Beginn der Reintegration erkannten Einschränkungen durch Abhängigkeiten der Konflikte zu Konflikten des
25 Typs HK1 und HK2 werden abhängig von der Lösung der Konflikte des Typs HK1 und HK2 während der Reintegration dynamisch geändert. So kommt es abhängig vom Typ der abhängigen Konflikte und der jeweiligen Lösungsentscheidung der createElement-Operation und deleteElement-Operation zu den nachfolgenden
30 Änderungen der Entscheidungseinschränkungen:

a) Die Annahme einer createElement-Operation (erste Entscheidung E1 zu einem Konflikt vom Typ HK1) verursacht bei allen von der Operation abhängigen Konflikten eine Rücksetzung der Entscheidungseinschränkungen, d.h. eine Erweiterung der
35 Entscheidungsmöglichkeiten.

b) Die Ablehnung einer deleteElement-Operation (zweite Entscheidung E2 zu einem Konflikt vom Typ HK2) verursacht

ebenfalls eine Rücksetzung der Entscheidungseinschränkungen, d.h. eine Erweiterung der Entscheidungsmöglichkeiten für die von diesem Konflikt abhängigen Konflikte.

5 c) Die Ablehnung einer createElement-Operation (zweite Entscheidung E2 zu einem Konflikt vom Typ HK1) führt zur Beibehaltung der bereits vorgenommenen Entscheidungseinschränkungen der von diesem Konflikt abhängigen Konflikte. Es ändern sich somit keine Entscheidungsmöglichkeiten.

10 d) Die Annahme einer deleteElement-Operation (erste Entscheidung E1 zu einem Konflikt vom Typ HK2) führt zur Beibehaltung der bereits vorgenommenen Entscheidungseinschränkungen der von diesem Konflikt abhängigen Konflikte. Es ändern sich somit keine Entscheidungsmöglichkeiten.

15 Eine einseitig erzeugte Anomalie kann über jeweils zwei vorhandene Konflikte in einer Historie erkannt werden. Dabei gibt es die Möglichkeiten der Konfliktpaare HK5Aa/HK5Ab, HK4A/HK3A, HK4A/HK5A. Bei einem Konflikt vom Typ HK5Aa ist die deleteBeziehung-Operation und bei einem Konflikt vom Typ
20 HK5Ab ist die createBeziehung-Operation an der Anomalie beteiligt.

Für Anomalien, die durch Veränderungen auf der Kopiedatenbank des zweiten Benutzers B entstehen, gelten die gleichen Möglichkeiten: HK5Ba/HK5Bb, HK4B/HK3B, HK4B/HK5B. Allen Konfliktpaaren ist dabei gemeinsam, daß es eine deleteBeziehung-Operation mit einem Identifier id1 als Quelldatenelement und einem Identifier id2 als Zieldatenelement in einem der beiden Konflikte gibt und die gleichen Identifier vertauscht als
30 Quelldatenelement und Zieldatenelement in einer createBeziehung-Operation des anderen Konfliktes auftreten.

Die Annahme des Konflikts der einseitigen Anomalie, wie oben beschrieben, mit der createBeziehung-Operation verhindert die
35 Ablehnung des Konflikts der einseitigen Anomalie mit der deleteBeziehung-Operation einer Ablehnung des Konflikts der einseitigen Anomalie mit der deleteBeziehung-Operation dage-

gen verhindert die Annahme des Konflikts der Anomalie mit der createBeziehung-Operation. Die Änderung einer der beiden Konflikte vermindert die Entscheidungsmöglichkeiten des anderen Konfliktes nicht.

5

Somit ergibt sich, daß eine einseitig erzeugte Anomalie gerichteter Beziehungen durch eine Ablehnung beider Konflikte, einer Annahme beider Konflikte, einer verschiedenen Änderung beider Konflikte oder einer Veränderung einer der Konflikte und einer Annahme oder Ablehnung des anderen Konflikts lösbar ist.

10

Eine beidseitig erzeugte Anomalie gerichteter Beziehungen ist durch die Entscheidung einer der beiden Konflikte und eine davon verschiedene Entscheidung des anderen Konfliktes lösbar.

15

Zur Vermeidung der Entstehung einer Anomalie (sogenannte Pseudo-Anomalie), wie oben beschrieben, sind folgende Einschränkungen der Entscheidungsmöglichkeiten vorgesehen:

20

a) Nach einer Annahme eines Konflikts mit der createBeziehung-Operation, die das gemeinsame Datenelement id1 als Zieldatenelement enthält (createBeziehung21) darf es für den Konflikt mit den zwei createBeziehung-Operationen, die das gemeinsame Datenelement als Quelldatenelement enthalten, keine Entscheidungsmöglichkeit der sechsten Entscheidungsmöglichkeit E6 mit einer Ersetzung des Zieldatenelements (idx oder idz) durch das Quelldatenelement id2(createBeziehung21) mehr geben.

25

b) Nach einer erfolgten sechsten Entscheidung E6 aufgrund der sechsten Entscheidungsmöglichkeit E6 und der Auswahl eines neuen Zieldatenelements id2 für den Konflikt mit den zwei createBeziehung-Operationen, die das gemeinsame Datenelement id1 als Quelldatenelement enthalten, darf keine Annahme des Konfliktes mit der createBeziehung-Operation mit dem gemeinsamen Datenelement id1 als Zieldatenelement (createBeziehung21) mehr möglich sein.

30

35

Wie oben beschrieben, werden im Rahmen dieses Verfahrens die Entscheidungsmöglichkeiten von Konflikten entsprechend deren Abhängigkeiten, Anomalien und Pseudo-Anomalien eingeschränkt.

5

Nach jeder Entscheidung zu einem Konflikt wird eine Änderung der Entscheidungsmöglichkeiten der von dem eben gelösten Konflikt abhängigen oder sich in einer gemeinsamen Anomalie bzw. Pseudo-Anomalie mit diesem Konflikt befindlichen Konflikte entsprechend den Abhängigkeiten, Anomalien und Pseudo-Anomalien vorgenommen.

10

Für jeden Konflikt wird, wie oben beschrieben, eine Entscheidung getroffen. Die Entscheidung kann auf unterschiedliche Weise erfolgen. Eine Übersicht über mögliche Entscheidungsvariationen ist in [1] zu finden.

15

Im Rahmen dieses Ausführungsbeispiels ist vorgesehen, daß eine Datenbank oder Kopiedatenbank als Referenzdatenbank angesehen wird, und der Abgleich gemäß der Referenzdatenbank erfolgt.

20

Es wird also, wie in Fig.1 durch eine rekursive Schleife über einem Überprüfungsschritt (Schritt 105) dargestellt ist, überprüft, ob noch ein Konflikt vorliegt und somit eine Entscheidung getroffen werden muß. Ist eine Entscheidung noch zu treffen, so wird diese getroffen. Sind keine Konflikte mehr vorhanden, so wird ein letzter Verfahrensschritt (Schritt 106) durchgeführt, das Abspeichern der reintegrierten Datenbank, welche keine Inkonsistenzen mehr aufweist.

25

30

Die konsistenzfreie Datenbank wird wieder an alle weiteren Rechner, die mit dem ersten Rechner 200 verbunden sind, übertragen (Schritt 107).

35

Damit besitzen alle Rechner eine konsistente Kopiedatenbank.

Im weiteren werden einige Alternativen zu dem oben beschriebenen Ausführungsbeispiel dargestellt:

5 Die Erkennung von Inkonsistenzen kann auch nach einer vorgebbaren Anzahl erfolgter Behebung einer Inkonsistenz durch die Suche nach einer weiteren Inkonsistenz erfolgen. Dies kann dahingehend erweitert werden, daß erneut nach jeder Behebung einer Inkonsistenz die Suche nach einer nächsten Inkonsistenz und deren Behebung erfolgt.

10

Es ist ferner möglich, daß durch den ersten Rechner gemäß dem oben dargestellten Verfahren eine Folge von Korrekturbefehlen (Korrektursequenzen) ermittelt wird, die jeweils dem Rechner, dessen Kopiedatenbank auf Inkonsistenzen hin überprüft wurde, 15 übermittelt wird und der jeweilige Rechner anhand der Korrektursequenz seine Kopiedatenbank an die Datenbank abgleicht.

20

Es ist ferner in einer alternativen Ausführungsform ebenfalls möglich, einem Benutzer oder mehreren Benutzern die Entscheidung zu überlassen, d.h. die Entscheidungsmöglichkeiten werden einem Benutzer auf dem Bildschirm dargestellt, und der Benutzer wählt über die Tastatur oder die Computermouse die von ihm gewünschte Entscheidung aus, die dann von dem Rechner durchgeführt wird.

In diesem Dokument ist folgende Veröffentlichung zitiert:

[1] DE 196 07 132 A1

Patentansprüche

1. Verfahren zur rechnergestützten Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht,
 - a) bei dem mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, definierten Konflikttypen zugeordnet ist,
 - b) bei dem jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
 - c) bei dem die Inkonsistenz unter Verwendung des Entscheidungssets behoben wird.
2. Verfahren nach Anspruch 1, bei dem mehrere Inkonsistenzen behoben werden.
3. Verfahren nach Anspruch 1 oder 2, bei dem jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mehrere Operationen des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
4. Verfahren nach einem der Ansprüche 1 bis 3, bei dem die Datenbankmenge mehrere Kopiedatenbanken der Datenbank aufweist.
5. Verfahren nach einem der Ansprüche 1 bis 4, bei dem vor der Behebung der Inkonsistenzen alle Inkonsistenzen und deren Abhängigkeiten voneinander, Anomalien und Pseudo-Anomalien ermittelt werden.
6. Verfahren nach einem der Ansprüche 1 bis 5,

bei dem das Entscheidungsset eines Konfliktes während des Verfahrens verändert wird.

7. Verfahren nach einem der Ansprüche 1 bis 6,
5 bei dem die Änderung des jeweiligen Entscheidungssets abhängig von Abhängigkeiten von Inkonsistenzen erfolgt.

8. Verfahren nach einem der Ansprüche 1 bis 7,
10 bei dem nach einer vorgebbaren Anzahl von behobenen Inkonsistenzen die Datenbankmenge auf weitere Inkonsistenzen und deren Abhängigkeiten, Anomalien und Pseudo-Anomalien hin untersucht wird.

9. Verfahren nach einem der Ansprüche 1 bis 8,
15 bei dem die Datenbankmenge eine objektorientierte Datenbank enthält.

10. Verfahren nach einem der Ansprüche 1 bis 9,
20 eingesetzt im Rahmen der objektorientierten Softwareentwicklung.

11. Verfahren nach einem der Ansprüche 1 bis 9,
eingesetzt im Rahmen der Erstellung eines strukturierten elektronischen Dokuments.

25 12. Anordnung zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht,
30

mit mindestens einem Prozessor, der derart eingerichtet ist, daß folgende Schritte durchführbar sind:

- 35 a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit de-

nen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben.

5

13. Anordnung nach Anspruch 12,
bei der der Prozessor derart eingerichtet ist, daß mehrere Inkonsistenzen behoben werden.

10

14. Anordnung nach Anspruch 12 oder 13,
bei der der Prozessor derart eingerichtet ist, daß jedem Konflikttyp ein Entscheidungsset zugeordnet ist, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mehrere Operationen des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,

15

15. Anordnung nach einem der Ansprüche 12 bis 14,
bei der der Prozessor derart eingerichtet ist, daß die Datenbankmenge mehrere Kopiedatenbanken der Datenbank aufweist.

20

16. Anordnung nach einem der Ansprüche 12 oder 15,
bei der der Prozessor derart eingerichtet ist, daß vor der Behebung der Inkonsistenzen alle Inkonsistenzen und deren Abhängigkeiten voneinander, Anomalien und Pseudo-Anomalien ermittelt werden.

25

17. Anordnung nach einem der Ansprüche 12 bis 16,
bei der der Prozessor derart eingerichtet ist, daß das Entscheidungsset eines Konfliktes während des Verfahrens verändert wird.

30

18. Anordnung nach einem der Ansprüche 12 bis 17,
bei der der Prozessor derart eingerichtet ist, daß die Änderung des jeweiligen Entscheidungssets abhängig von Abhängigkeiten von Inkonsistenzen erfolgt.

35

19. Anordnung nach einem der Ansprüche 12 bis 18,

bei der der Prozessor derart eingerichtet ist, daß nach einer vorgebbaren Anzahl von behobenen Inkonsistenzen die Datenbankmenge auf weitere Inkonsistenzen und deren Abhängigkeiten, Anomalien und Pseudo-Anomalien hin untersucht wird.

5

20. Anordnung nach einem der Ansprüche 12 bis 19, bei der der Prozessor derart eingerichtet ist, daß die Datenbankmenge eine objektorientierte Datenbank enthält.

10 21. Anordnung nach einem der Ansprüche 12 bis 20, eingesetzt im Rahmen der objektorientierten Softwareentwicklung.

15 22. Anordnung nach einem der Ansprüche 12 bis 21, eingesetzt im Rahmen der Erstellung eines strukturierten elektronischen Dokuments.

23. Satz mehrerer Anordnungen zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Datenbank und/oder der Kopiedatenbank entsteht,
20 bei dem jede Anordnung mindestens einen Prozessor aufweist, der derart eingerichtet ist, daß folgende Schritte durchführbar sind:
25

- a) mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, ist definierten Konflikttypen zugeordnet,
 - b) jedem Konflikttyp ist ein Entscheidungsset zugeordnet,
30 mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch mindestens eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann,
 - c) die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben, und
- 35 bei dem die Anordnungen miteinander koppelbar sind.

Zusammenfassung

Verfahren, Anordnung und Satz mehrerer Anordnungen zur Behebung von Inkonsistenzen in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist

Zur Behebung mindestens einer Inkonsistenz in einer Datenbankmenge, die eine Datenbank sowie mindestens eine Kopiedatenbank der Datenbank aufweist, welche Inkonsistenz durch Änderung der Daten in der Datenbank oder in der Kopiedatenbank entsteht, ist mindestens ein Teil der Operationen, die eine Inkonsistenz erzeugen können, definierten Konflikttypen zugeordnet. Jedem Konflikttyp ist ein Entscheidungsset zugeordnet, mit dem mögliche Entscheidungen angegeben werden, mit denen eine durch eine Operation des jeweiligen Konflikttyps erzeugte Inkonsistenz behoben werden kann. Die Inkonsistenz wird unter Verwendung des Entscheidungssets behoben. Eine fehlerfreie Behebung von Inkonsistenzen wird durch Anpassung des Entscheidungssets an die jeweilige Situation gewährleistet.

FIG 1

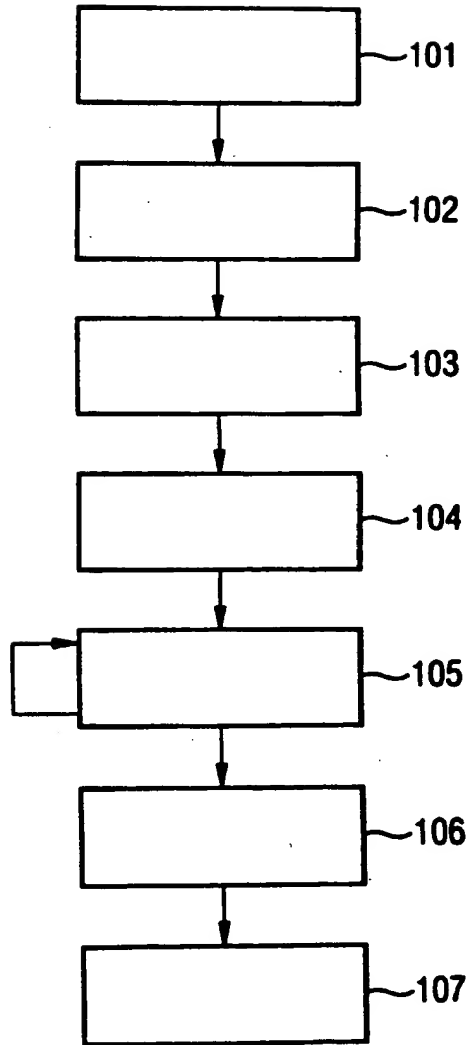


FIG 2

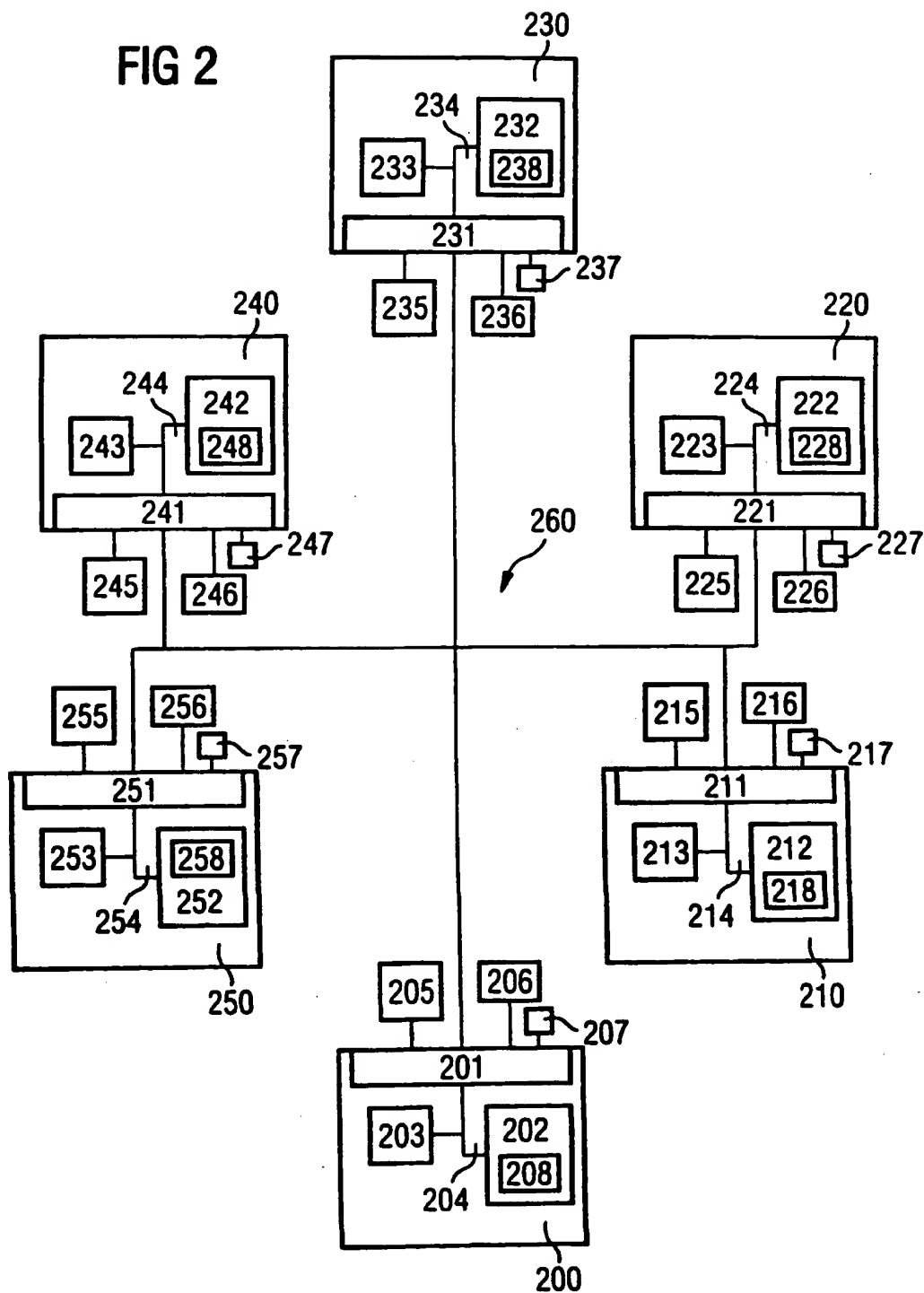


FIG 3

